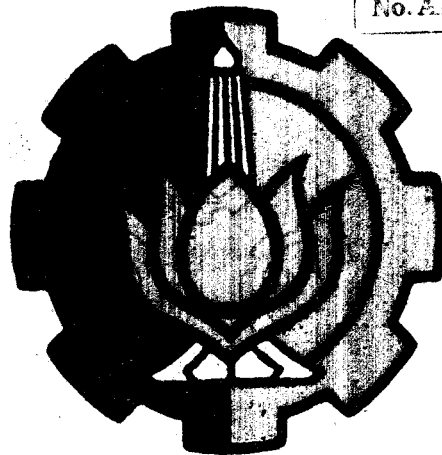


6168/TS/H/94 ✓

**ALAT UNTUK MENENTUKAN KANDUNGAN AIR  
DALAM BATUBARA YANG DIINTERFACEKAN  
PADA IBM PC-XT**

|                     |             |
|---------------------|-------------|
| PERPUSTAKAAN<br>ITS |             |
| Tgl. Terima         | 20 OCT 1993 |
| Terima Dari         | H.          |
| No. Agenda Perp.    | 1370 /TA.   |



RSE  
621.3981  
Kar  
a-1  
1993

*Oleh :*

**I KOMANG WIDNYANA KARANG**

**NRP. 2882200945**

**JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
SURABAYA  
1993**

# **ALAT UNTUK MENENTUKAN KANDUNGAN AIR DALAM BATUBARA YANG DIINTERFACEKAN PADA IBM PC-XT**

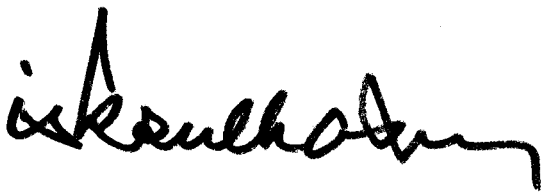
## **TUGAS AKHIR**

**Diajukan Guna Memenuhi Sebagian Persyaratan  
Untuk Memperoleh Gelar Sarjana Teknik Elektro  
Pada**

**Bidang Studi Elektronika  
Jurusan Teknik Elektro  
Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember  
S u r a b a y a**

**Mengetahui/Menyetujui**

**Dosen Pembimbing I**



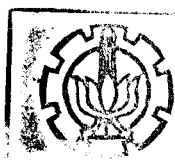
**(Ir. ISKANDAR ZULKARNAIN)**

**Dosen Pembimbing II**



**(Ir. DJOKO PURWANTO)**

**S U R A B A Y A  
AGUSTUS, 1993**



**MILIK PERPUSTAKAAN  
INSTITUT TEKNOLOGI  
SEPULUH - NOPEMBER**

## ABSTRAK

Menentukan Kandungan air dalam batubara sangat penting karena mempengaruhi nilai kalor dan performance bahan bakar. Penentuan kandungan air dilakukan dengan mengukur berat yang hilang dari sampel batubara saat dipanaskan pada temperatur kerja. Dalam tugas akhir ini dibuat interface ke IBM PC-XT yang digunakan untuk mengontrol pengukuran berat dan pemanas.

Pengukuran berat ini dilakukan dengan merubah gaya berat yang bekerja menjadi tegangan listrik dengan menggunakan strain gauge dalam rangkaian jembatan wheatstone. Beda tegangan yang dihasilkan rangkaian jembatan dikuatkan dan dikonversikan ke digital untuk kemudian diolah pada IBM PC-XT. Dan untuk mendapatkan temperatur kerja dilakukan dengan cara mengontrol penyalaan pemanas. Pengontrolan pemanas ini dilakukan secara proporsional dengan bantuan IBM PC-XT. Hasil pengukuran berat dan pengontrolan pemanas tersebut ditampilkan pada monitor dan selanjutnya dapat disimpan pada diskette atau dicetak pada printer.

## KATA PENGANTAR

Berkat Rakhmat Tuhan Yang Maha Pengasih dan Maha Penyayang, akhirnya Tugas Akhir ini yang merupakan syarat dalam menyelesaikan studi strata satu pada Jurusan Teknik Elektro, Fakultas Teknik Industri, Institut Teknologi Sepuluh Nopember Surabaya, dapat diselesaikan dengan baik.

Penyelesaian Tugas Akhir ini berdasarkan pada teori-teori yang penulis dapatkan selama kuliah, buku-buku referensi, bimbingan dosen dan pihak-pihak lain yang ikut memberikan dukungan dan saran.

Ucapan terima kasih yang sebesar-besarnya penulis sampaikan kepada :

1. Bapak Ir. Iskandar Zulkarnain dan Bapak Ir. Djoko Purwanto, selaku dosen pembimbing yang banyak memberikan bimbingan, saran serta dorongan dalam penyelesaian Tugas Akhir ini.
2. Bapak Ir. Totok Mujiono, M.Sc, selaku dosen wali penulis.
3. Bapak Ir. Soetikno, selaku koordinator bidang studi Elektronika.
4. Bapak Katjuk Astrowulan, MSEE, selaku ketua jurusan Teknik Elektro FTI - ITS.

5. Ir. I Putu Sutrisna, yang sekaligus sebagai kakak yang telah membimbing dan membantu penyelesaian Tugas Akhir ini.

6. Semua pihak yang tak dapat penulis sebutkan satu persatu, telah banyak memberikan pertimbangan dan dorongan sehingga Tugas Akhir ini bisa terselesaikan.

Akhirnya semoga apa yang tertuang dalam tulisan ini ada manfaatnya.

Surabaya, Juli 1993

I Komang Widnyana Karang

## DAFTAR ISI

|   | Halaman |
|---|---------|
| ABSTRAK .....                                 | i       |
| KATA PENGANTAR .....                          | ii      |
| DAFTAR ISI .....                              | iv      |
| DAFTAR GAMBAR .....                           | vii     |
| DAFTAR TABEL .....                            | ix      |
| BAB I. PENDAHULUAN .....                      | 1       |
| I.1. LATAR BELAKANG .....                     | 1       |
| I.2. PERMASALAHAN .....                       | 2       |
| I.3. TUJUAN .....                             | 2       |
| I.4. METODOLOGI .....                         | 2       |
| I.5. SISTEMATIKA .....                        | 3       |
| I.6. RELEVANSI .....                          | 4       |
| BAB II. TEORI PENUNJANG .....                 |         |
| II.1. BATUBARA DAN SIFAT-SIFATNYA .....       | 5       |
| II.2. PROGRAMMABLE PERIPHERAL INTERFACE ..... | 8       |
| II.2.1. PPI INTEL 8255 .....                  | 9       |
| II.2.2. MODE KERJA PPI 8255 .....             | 11      |
| II.3. PROGRAMMABLE INTERVAL TIMER .....       | 14      |
| II.3.1. PIT INTEL 8253 .....                  | 14      |
| II.4. ANALOG TO DIGITAL CONVERTER .....       | 21      |
| II.5. SENSOR TEMPERATUR .....                 | 24      |
| II.5.1. SENSOR IC TEMPERATUR .....            | 26      |

|   |    |
|---|----|
| II.6. SENSOR BERAT .....  | 28 |
| II.6.1. KONSEP STRAIN GAUGE .....   | 31 |
| II.6.2. PRINSIP KERJA STRAIN GAUGE .....                                    | 32 |
| II.6.3. SENSITIVITAS REGANGAN .....   | 33 |
| II.6.4. FAKTOR GAUGE .....  | 35 |
| II.6.5. SUSUNAN STRAIN GAUGE .....  | 36 |
| II.6.6. PEMASANGAN STRAIN GAUGE<br>PADA JEMBATAN WHEATSTONE .....           | 37 |
| II.6.7. MENYEIMBANGAN SEBUAH<br>JEMBATAN WHEATSTONE .....                   | 39 |
| BAB III. PERENCANAAN SISTEM PENENTUAN KANDUNGAN AIR<br>DALAM BATUBARA ..... | 42 |
| BAB IV. PERENCANAAN PERANGKAT ELEKTRONIS                                    |    |
| IV.1. PERENCANAAN HARDWARE .....  | 47 |
| IV.1.1. MODUL INTERFACE .....   | 49 |
| IV.1.2. MODUL PENGATUR PEMANAS .....  | 57 |
| IV.1.3. MODUL PENGGERAK KIPAS PENDINGIN ..                                  | 63 |
| IV.1.4. MODUL SENSOR .....  | 64 |
| IV.1.5. MODUL ADC .....   | 76 |
| IV.2. PERENCANAAN SOFTWARE .....  | 81 |
| BAB V. PENGUJIAN DAN PENGUKURAN PERALATAN                                   |    |
| V.1. PENGUJIAN SENSOR TEMPERATUR .....                                      | 88 |
| V.2. PENGUJIAN SENSOR BERAT .....   | 93 |
| V.3. PENGUJIAN TERHADAP RANGKAIAN<br>UMPAN-BALIK PROPOSIONAL .....          | 97 |

|   |     |
|---|-----|
| V.4. PENENTUAN KANDUNGAN AIR DALAM BATUBARA ... | 97  |
| BAB VI. KESIMPULAN                              |     |
| VI.1. KESIMPULAN .....                          | 100 |
| VI.2. SARAN .....                               | 101 |
| DAFTAR PUSTAKA                                  |     |
| LAMPIRAN  |     |



## DAFTAR GAMBAR

|   | Halaman |
|---|---------|
| 2.1. SIFAT-SIFAT UNTUK RANGKING BERBEDA DARI BATUBARA .   | 7       |
| 2.2. BLOK DIAGRAM PPI 8255 DAN KONFIGURASI PIN-PINNYA .   | 10      |
| 2.3. FORMAT CONTROL WORD .....  | 12      |
| 2.4. MODE 0 (BASIC INPUT-OUTPUT) PPI 8255 .....   | 13      |
| 2.5. KONFIGURASI PIN DAN BLOK DIAGRAM PIT 8253 .....  | 15      |
| 2.6. FORMAT CONTROL WORD .....  | 18      |
| 2.7. DIFINISI CONTROL WORD REGISTER .....   | 19      |
| 2.8. TIMING DIAGRAM MODE 1 PIT 8253 .....   | 20      |
| 2.9. ADC TIPE SUCCESSIVE APPROXIMATION .....  | 23      |
| 2.10. PERBANDINGAN SENSOR TEMPERATUR .....  | 25      |
| 2.11. IC SENSOR TEMPERATUR LM 335 .....   | 28      |
| 2.12. PENGUKURAN GAYA BERAT .....   | 29      |
| 2.13. KURVA STRESS DAN STRAIN .....   | 31      |
| 2.14. STRAIN GAUGE SATU SUMBU .....   | 37      |
| 2.15. PEMASANGAN STRAIN GAUGE .....   | 38      |
| 2.16. JEMBATAN DENGAN RANGKAIAN PENYEIMBANG .....   | 39      |
| 2.20. DIAGRAM ALIR SISTEM PENENTUAN KANDUNGAN AIR<br>DALAM BATUBARA .....                       | 43      |
| 2.21. BLOK DIAGRAM DARI PERALATAN YANG DIRANCANG BERBASIS<br>PADA MIKROKOMPUTER IBM PC-XT ..... | 45      |
| 4.1. BLOK DIAGRAM PERANGKAT ELEKTRONIS DARI PERALATAN<br>YANG DIRENCANAKAN .....                | 48      |

|       |   |    |
|-------|---|----|
| 4.2.  | MODUL INTERFACE .....   | 50 |
| 4.3.  | RANGKAIAN BUFFER DATA BUS .....                                   | 51 |
| 4.4.  | RANGKAIAN DECODER .....   | 54 |
| 4.5.  | RANGKAIAN PPI 8255 .....  | 55 |
| 4.6.  | RANGKAIAN PIT 8253 .....  | 57 |
| 4.7.  | RANGKAIAN DAN SINYAL SINKRONISASI .....                           | 58 |
| 4.8.  | PIT 8253 SEBAGAI RANGKAIAN PENGONTROL PEMANAS ....                | 60 |
| 4.9.  | RANGKAIAN PENGGERAK KIPAS PENDINGIN .....                         | 64 |
| 4.10. | RANGKAIAN PENGKONDISI SINYAL TEMPERATUR .....                     | 65 |
| 4.11. | PEMASANGAN STRAIN GAUGE PADA SPECIMEN .....                       | 69 |
| 4.12. | PEMASANGAN STRAIN GAUGE PADA JEMBATAN WHEATSTONE .                | 70 |
| 4.13. | RANGKAIAN PENGUAT INSTRUMENTASI .....                             | 74 |
| 4.14. | RANGKAIAN TEGANGAN 10 V DENGAN TRANSISTOR<br>PENDORONG ARUS ..... | 75 |
| 4.15. | ADC 0808 .....  | 76 |
| 4.16. | DIAGRAM WAKTU ADC 0808 .....                                      | 77 |
| 4.17. | RANGKAIAN ADC 0808 .....  | 78 |
| 4.18. | RANGKAIAN CLOCK ADC 0808 .....                                    | 80 |
| 4.19. | RANGKAIAN TEGANGAN REFERENSI 1,28 V .....                         | 81 |
| 4.20. | FLOWCHART DARI PROGRAM YANG DIBUAT SECARA<br>GARIS BESAR .....    | 82 |
| 4.21. | DIAGRAM KONTROL PROPORSIONAL YANG DIBUAT .....                    | 85 |
| 5.1.  | SINYAL SINKRONISASI .....   | 98 |
| 5.2.  | PROTOTYPE PERALATAN YANG DIBUAT .....                             | 98 |
| 5.3.  | HASIL PENGUKURAN DICETAK PADA PRINTER .....                       | 99 |

## DAFTAR TABEL

|  | Halaman |
|--|---------|
| 2.1. OPERASI PIT .....   | 17      |
| 2.2. DAFTAR PEMILIHAN SENSOR TEMPERATUR .....                                    | 27      |
| 4.1. TABEL KEBENARAN PENGKODEAN I/O PORT .....                                   | 53      |
| 4.2. TABEL KEBENARAN 74LS138 .....   | 54      |
| 5.1. HASIL PENGUJIAN SENSOR TEMPERATUR .....                                     | 89      |
| 5.2. HASIL PENGUKURAN SENSOR TEMPERATUR SETELAH<br>DIISOLASI .....               | 90      |
| 5.3. HASIL PENGUJIAN SENSOR TEMPERATUR DENGAN<br>PENDEKATAN REGRESI LINIER ..... | 92      |
| 5.4. HASIL PENGUKURAN TEGANGAN OUTPUT PENGKONDISI<br>SINYAL .....                | 93      |
| 5.5. HASIL PENGUJIAN SENSOR BERAT .....  | 96      |
| 5.6. HASIL PENENTUAN KANDUNGAN AIR DALAM BATUBARA .....                          | 97      |

# BAB I

## PENDAHULUAN

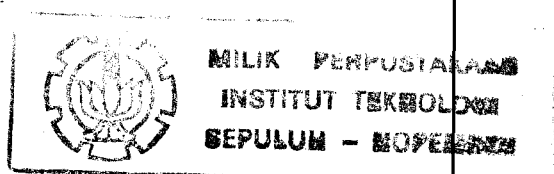
---

### I.1. LATAR BELAKANG

Menentukan kandungan air (moisture) dalam batubara sangat penting karena mempengaruhi nilai kalor (calorific value) dan performance bahan bakar tersebut terhadap proses pembakaran dan proses konversi batubara menjadi bahan bakar gas atau bahan bakar cair.

Penentuan kandungan air dalam batubara dilakukan dengan cara mengukur berat yang hilang dari sampel batubara saat dipanaskan pada temperatur berkisar antara 104 °C sampai dengan 110 °C. Kandungan air dalam batubara ini dinyatakan dalam persen berat.

Penentuan kandungan air dalam batubara ini secara konvensional membutuhkan peran manusia yang cukup dominan karena sebelum dan sesudah pemanasan dilakukan penimbangan sampel berulang-ulang sampai menghasilkan pengukuran berat yang konstan dan pembacaan hasil pengukurannya secara analog. Kemudian untuk memperoleh temperatur pemanasan antara 104 °C sampai dengan 110 °C dilakukan dengan penyalaan pemanas secara manual. Penentuan kandungan air secara konvensional ini kurang efektif, oleh karena itu direncanakan alat untuk menentukan kandungan air dalam batubara yang diinterfacekan pada



mikrokomputer IBM PC-XT. Dengan peralatan ini akan memudahkan pembacaan, mengurangi peran dari manusia dan waktu yang dibutuhkan relatif singkat.

## I.2. PERMASALAHAN

Kandungan air dalam batubara sangat mempengaruhi nilai kalor (calorific value) dan performance bahan bakar, sehingga membutuhkan penentuan kandungan air dalam batubara.

Penentuan kandungan air dalam batubara secara konvensional kurang memuaskan oleh karena membutuhkan peran manusia yang cukup dominan dan pembacaannya secara analog. Sehingga waktu yang diperlukan relatif lama dan kesalahan yang dihasilkan pembacaan hasil pengukuran ini masih sangat besar.

## I.3. TUJUAN

Tujuan pembuatan tugas akhir ini adalah merancang dan membuat alat untuk menentukan kandungan air dalam batubara yang diinterfacekan pada mikrokomputer IBM PC-XT.

## I.4. METODOLOGI

Untuk mencapai tujuan yang telah disebutkan di atas dilakukan langkah-langkah sebagai berikut :

- Studi literatur yang berhubungan dengan masalah yang dihadapi. Literatur ini adalah mengenai batubara, perangkat pendukung interface, analog to digital converter, sensor

temperatur dan strain gauge.

- Merencanakan perangkat keras yang meliputi rangkaian pengkondisi sinyal, rangkaian pengontrol pemanas, rangkaian penggerak kipas penghisap, rangkaian pengubah analog ke digital dan rangkaian PPI.
- Membuat perangkat keras dan perangkat lunak sesuai dengan perencanaan yang telah dibuat. Dilanjutkan dengan pengujian alat.
- Menyimpulkan perancangan dan pembuatan alat sesuai dengan hasil pengujian. Memberikan saran-saran untuk pengembangan alat ini lebih lanjut.
- Dari langkah-langkah diatas disusun buku laporan tugas akhir ini.

#### I.5. SISTEMATIKA

Sistematika dari laporan tugas akhir ini disusun sebagai berikut :

- BAB I merupakan bab pendahuluan berisi tentang latar belakang, permasalahan, tujuan, metodologi, sistematika dan relevansi.
- BAB II membahas tentang teori penunjang batubara dan sifat-sifatnya yang berkaitan dengan kandungan airnya dan komponen-komponen yang berhubungan dengan alat yang direncanakan antara lain : PPI 8255, PIT 8253, analog to digital converter (ADC), sensor temperatur dan strain gauge.

## BAB II

### TEORI PENUNJANG

---

Dalam bab ini dibahas tentang teori-teori yang berhubungan dengan peralatan yang akan direncanakan. Perangkat yang direncanakan akan bekerja dengan bantuan mikrokomputer IBM PC-XT.

Secara keseluruhan pada bab ini akan menguraikan tentang batubara, programmable peripheral interface (PPI), programmable interval timer (PIT), analog to digital converter (ADC), sensor temperatur dan strain gauge. Teori mengenai batubara tidak dibahas secara keseluruhan, namun hanya dijelaskan sifat-sifat dari batubara yang berkaitan dengan kandungan airnya.

#### II.1. BATUBARA DAN SIFAT-SIFATNYA

Batubara berasal dari sisa-sisa tumbuh-tumbuhan yang membusuk dan mengalami proses dekomposisi selama berjuta-juta tahun. Bentuk awal dari batubara adalah *peat* (gambut), terbentuk akibat aktifitas bakteri dan reaksi kimia pada sisa-sisa tumbuhan. Kemudian akibat gaya yang ditimbulkan oleh panas, tekanan dan gejala fisika yang lain merubah gambut menjadi batubara dalam berbagai rangking seperti sekarang ini. Disebabkan derajat perubahan yang berbeda-beda selama proses

ini maka batubara bukan merupakan zat yang homogen sehingga tidak ada batubara yang benar-benar sama.

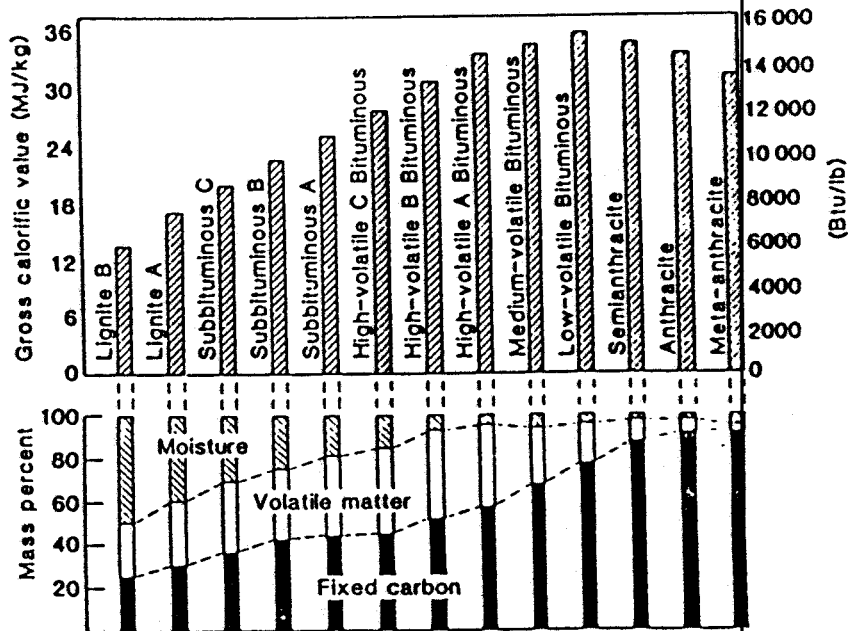
Ranking batubara menunjukkan tingkat perubahan kimia yang terjadi dalam batubara. Batubara ranking terendah yaitu *lignite*, kemudian diikuti oleh batubara *subbituminous*, *bituminous* dan *anthracite*. Umumnya semakin rendah ranking batubara maka nilai kalor (calorific value) semakin kecil dan kandungan air semakin tinggi.

Analisa proksimat yang didefinisikan sebagai fraksi moisture (kandungan air), volatile matter, ash dan fixed carbon merupakan sifat yang penting dari batubara yang harus diketahui karakteristiknya dalam proses konversi. Gambar 2.1 menunjukkan sifat-sifat tersebut untuk ranking berbeda.

Penentuan kandungan air batubara, disamping penentuan kandungan fixed carbon, volatile matter, dan ash juga nilai kalor sangat menentukan performance bahan bakar tersebut dalam proses pembakaran dan proses konversi batubara menjadi bahan bakar cair/gas.

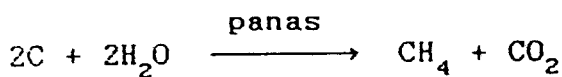
Dalam proses pembakaran batubara, kandungan air batubara perlu ditentukan dengan teliti dan tepat, karena kandungan air ini mempengaruhi kesetimbangan panas dari ruang bakar (furnace). Kandungan air semakin tinggi akan mengurangi karbon dalam batubara dan dapat mengakibatkan ketidakstabilan pengoperasian furnace, kecuali terdapat kompensasi dari penurunan karbon tersebut.





Gambar 2.1. Sifat-sifat untuk Rangkaian Berbeda dari Batubara<sup>1)</sup>

Pada proses konversi batubara menjadi bahan bakar gas (gasifikasi) seperti metana ( $\text{CH}_4$ ) perlu ditambah unsur hidrogen (H). Unsur hidrogen biasanya jarang ditemukan secara tersendiri di alam, akan tetapi terdapat secara melimpah pada air ( $\text{H}_2\text{O}$ ). Reaksi kimia dari proses gasifikasi batubara dengan uap air adalah sebagai berikut :



<sup>1)</sup> Ronald F. Probstein and R. Edwin Hicks, Synthetic Fuels, 1-st, McGraw-Hill International Book Company, 1982, hlm. 13.

Penentuan kandungan air batubara dalam proses ini perlu untuk menentukan jumlah dari uap air yang ditambahkan dalam proses.

## II.2. PRORAMMABLE PERIPHERAL INTERFACE (PPI)

Agar suatu peralatan luar (peralatan tambahan) dapat berhubungan dengan suatu sistem mikroprosesor atau mikrokomputer, maka diperlukan suatu teknik khusus untuk menghubungkan keduanya. Teknik ini disebut teknik interfacing. Dalam teknik interfacing dikenal 2 macam metode, yaitu serial interfacing dan paralel interfacing, tetapi di lain pihak paralel interfacing memerlukan jumlah kabel penghubung lebih banyak. Karena sifat ini, serial interfacing umumnya digunakan pada teknik transmisi jarak jauh, sedangkan paralel interfacing digunakan pada transmisi jarak dekat. Dalam paralel interfacing dikenal paralel input dan output interfacing dimana data ditransfer dari peralatan ke sistem mikrokomputer dan sebaliknya dari sistem mikrokomputer ke peralatan (output).

Untuk melakukan kedua fungsi tersebut, diperlukan IC interface yang sering disebut Programmable Peripheral Interface (PPI). Ada berbagai macam PPI yang diproduksi oleh berbagai pabrik. Namun diantara IC-IC PPI tersebut, IC PPI 8255 buatan Intel merupakan IC PPI yang paling banyak dipakai (termasuk dalam sistem board IBM PC). Karena itu dalam perencanaan pembuatan alat disini digunakan IC PPI 8255.

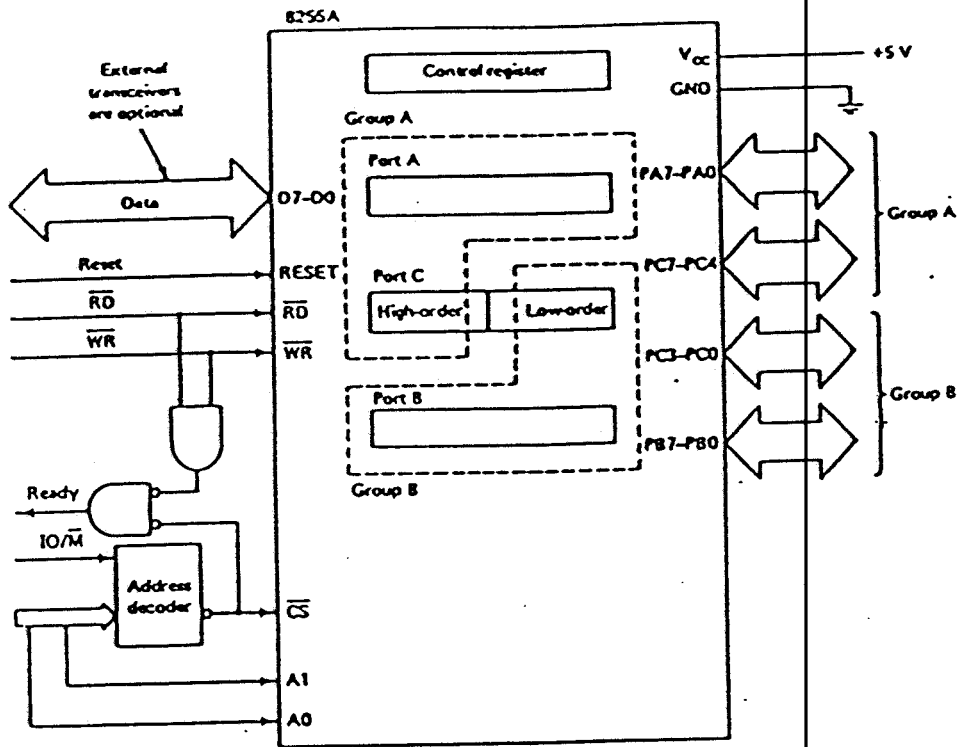
### II.2.1. PPI INTEL 8255

IC PPI 8255 adalah IC LSI yang didesain untuk mempermudah implementasi paralel I/O pada mikroprosesor 8088. IC ini memberikan interface secara paralel yang fleksibel, mencakup kemudahan operasi-operasi secara single bit, 4 bit serta 8 bit input dan output port. Selain itu IC ini juga mempunyai keunggulan yaitu input yang sensitif, output yang di-latch, strobe input dan output dan strobe input/output yang bidirectional. Fasilitas-fasilitas ini dikendalikan melalui perangkat lunak.

Gambar 2.2 menunjukkan blok diagram PPI Intel 8255. Sisi sebelah kiri dari diagram blok menunjukkan interfacing ke mikroprosesor. Terlihat adanya 8 bit data bus yang bi-directional ( $D_0$  sampai  $D_7$ ) dan sinyal-sinyal  $A_0$ ,  $A_1$ ,  $\overline{RD}$ ,  $\overline{WR}$ , RESET dan  $\overline{CS}$ .

Melalui data bus dapat ditransfer data, instruksi atau informasi status. Data ini ditransfer bila mikroprosesor 8088 melakukan siklus baca atau siklus tulis. Ini dilakukan oleh oleh.sinyal  $\overline{RD}$  dan  $\overline{WR}$ .

Register sumber atau tujuan dalam PPI 8255 dipilih oleh register select code (2 bit). Mikroprosesor 8088 harus memberikan kode ini ke input-input  $A_1$  dan  $A_0$  dari PPI 8255. Register port A, port B dan port C sesuai dengan kode  $A_1A_0 = 00$ ,  $A_1A_0 = 01$  dan  $A_1A_0 = 10$ .



Gambar 2.2. Blok Diagram PPI 8255 dan Konfigurasi pin-pinnya<sup>2)</sup>

Sinyal input  $\overline{CS}$  digunakan untuk memilih IC PPI ini.  $\overline{CS}$  ini harus berlogika '0' (low) selama operasi baca atau tulis ke PPI.

Input Reset digunakan untuk menginisialisasi IC. Bila pin ini dibuat high, maka semua register internal dari IC ini akan di-clear. Selain itu, inisialisasi ini juga mengkon-

<sup>2)</sup> Yu Cheng Liu, Glenn A. Gibson, Microcomputer Systems: 8086/8088 Family, Prentice-Hall International, 1986, hlm. 372.

figurasikan semua I/O port menjadi mode operasi input. Sisi kanan dari blok diagram menunjukkan 3 buah I/O port, masing-masing berukuran 1 byte. Port-port ini dinamai port A, port B dan port C. Port-port ini dapat dikonfigurasi untuk operasi input dan output, Ini memberikan total 24 jalur I/O. Sub-bab dibawah ini menerangkan pengaturan mode kerja 8255 dan operasi yang terjadi.

### II.2.2. MODE KERJA PPI 8255

Mode kerja dari PPI 8255 ditentukan melalui perangkat lunak. Untuk itu IC ini dilengkapi dengan kontrol register 8 bit. Pada register ini dapat dituliskan logika '0' atau '1' untuk mengkonfigurasi masing-masing port sebagai input atau output, dan menentukan mode operasi. Register kontrol ini merupakan write-only register, dan isinya diubah melalui perangkat lunak dengan melakukan siklus tulis ke 8255 dan  $A_1A_0 = 11$ .

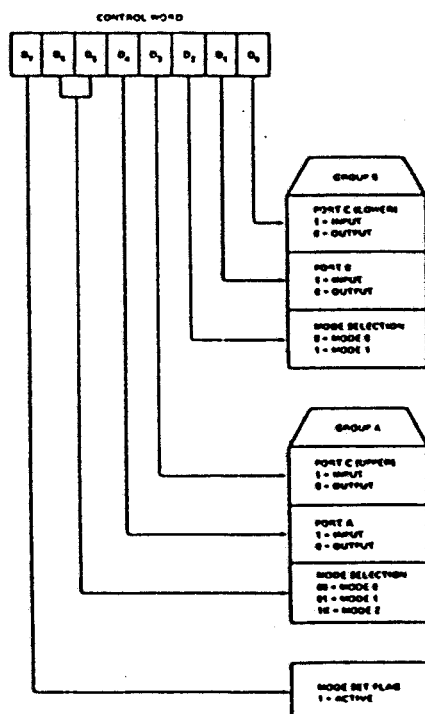
Gambar 2.3 menunjukkan bit-bit dari register kontrol dan fungsinya. Disini terlihat bit  $D_0$  sampai  $D_2$  berkorespondensi dengan grup B dari gambar 2.2. Bit  $D_0$  mengatur 4 bit dari port C lower sebagai input atau output. Logika 1 pada  $D_0$  menjadikan input dan logika '0' menjadikannya output.

Bit  $D_1$  mengatur port B (8 bit), logika '1' pada bit ini menjadikan input dan logika '0' menjadikannya output.

Bit  $D_2$  menentukan mode kerja port B (8 bit) dan port C lower (4 bit). Ada 2 mode operasi yang dapat dipilih, yaitu

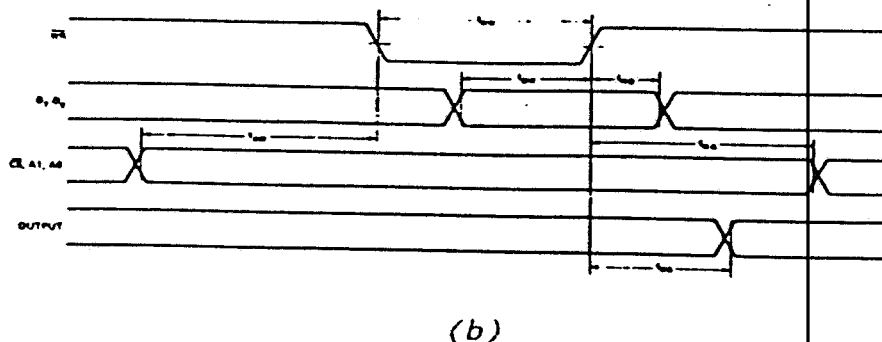
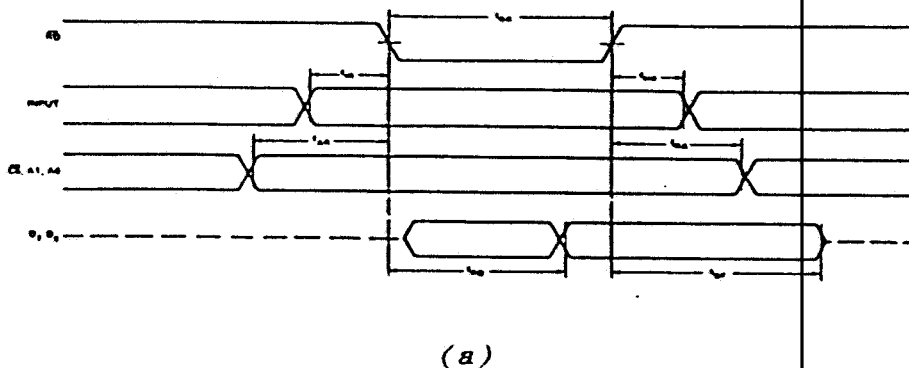
mode 0 dan mode 1. Logika '0' pada bit ini menjadikan mode 0 dan logika '1' menjadikannya mode 1.

Bit  $D_3$  sampai  $D_6$  berkorespondensi dengan grup A dari Gambar 2.2. Bit  $D_3$  dan  $D_4$  menentukan operasi kerja dari port C upper (4 bit) dan port A (8 bit), logika '1' menjadikan port-port ini sebagai input dan logika '0' menjadikannya sebagai output.  $D_5$  dan  $D_6$  berfungsi untuk memilih salah satu dari 3 mode operasi, yaitu mode 0, mode 1 dan mode 2.



Gambar 2.3. *Format Control Word*<sup>3)</sup>

<sup>3)</sup> J.P.M. Steeman, Data Sheet Book 2, PT. Elex Media Komputindo, Jakarta, 1988, hlm. 242.



Gambar 2.4. Mode 0 (Basic Input-Output)<sup>4)</sup>

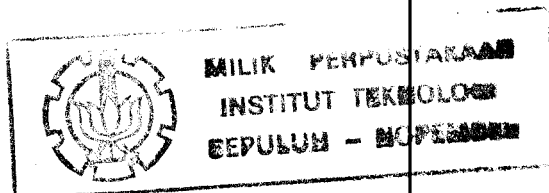
(a) Basic Input.

(b) Basic Output.

Bit  $D_7$  adalah mode set flag, bit ini harus berlogika '1' (aktif) bila mode operasi akan diubah.

Mengingat pada perencanaan peralatan hanya digunakan 1 mode saja, yaitu mode 0, maka hanya akan dibahas operasi mode 0 saja.

<sup>4)</sup> Ibid, hlm. 243.



### MODE 0 (BASIC INPUT OUTPUT)

Bila suatu grup berada dalam mode 0, grup itu dibagi menjadi 2 bagian. Untuk grup A kedua bagian ini adalah port A dan port C upper, sedangkan untuk grup B adalah port B dan port C lower.

Mode 0 ini digunakan untuk operasi input atau output yang sederhana yang terdiri dari 3 port. Tidak ada handshaking dan data hanya dibaca dan ditulis dari port-port tersebut. Gambar 2.4 menunjukkan timing diagram siklus baca dan tulis dari mode 0.

### II.3. PROGRAMMABLE INTERVAL TIMER (PIT 8253)

Dalam mikrokomputer banyak dipakai PIT Intel 8253 sebagai peralatan pencacah yang dapat diprogram melalui instruksi sesuai dengan besarnya frekuensi yang dibutuhkan. Selain sebagai pembagi frekuensi yang dapat diprogram, komponen ini juga mempunyai banyak fungsi yang dapat dipakai. Karena dalam tugas akhir ini menggunakan PIT Intel 8253, maka komponen ini akan dibahas lebih lanjut.

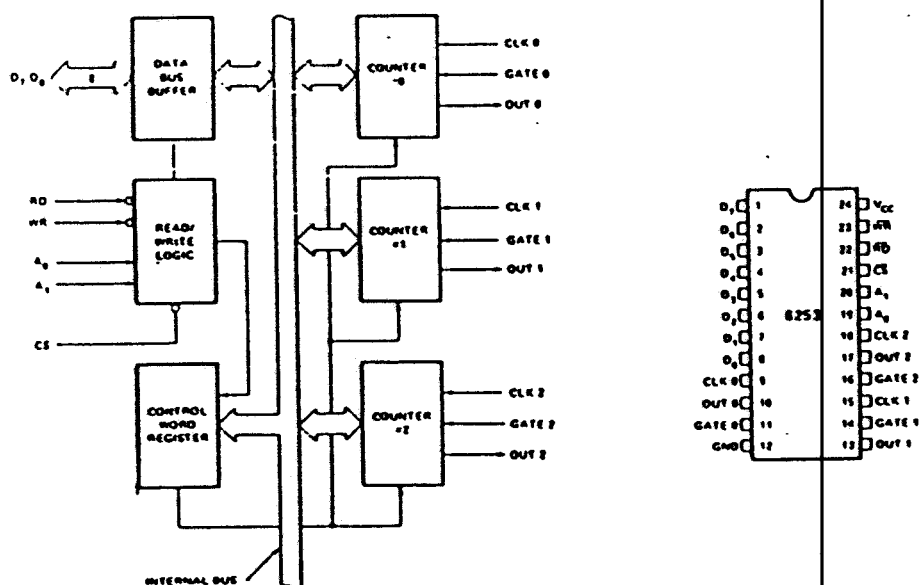
#### II.3.1. PIT INTEL 8253

PIT Intel 8253 adalah suatu peripheral LSI yang didesain untuk memudahkan implementasi fungsi-fungsi pewaktu dan counter. PIT 8253 dapat diprogram sehingga dapat bekerja sendiri tanpa mengganggu kerja dari CPU, PIT 8253 ini terdiri dari 3 pasang timing logic. Tiap pasang timing logic dapat



diprogram secara terpisah sebagai interval timer, pembangkit pulsa one shot, pembangkit gelombang kotak atau rate generator. Gambar 2.5 menunjukkan konfigurasi pin dan blok diagram PIT 8253.

PIT 8253 terdiri dari 6 buah blok yaitu : counter 0, counter 1, counter 2, data bus buffer, control word register dan read/write logic.



Gambar 2.5. Konfigurasi Pin dan Blok Diagram  
PIT 8253<sup>5)</sup>

5) Ibid, hlm. 231.

Antara ketiga counter, control word register dan data buffer dihubungkan oleh satu terminal bus yang berfungsi untuk memasukkan atau mengeluarkan data. Semuanya bidirectional kecuali yang menuju control word register hanya satu arah.

Counter 0, 1 dan 2 ini simetris dan independent artinya tidak saling mempengaruhi satu sama lainnya. Setiap counter terdiri dari 16 bit counter dan memiliki sinyal masukan GATE dan CLK serta sinyal keluaran OUT. Ketiga counter ini dapat dioperasikan pada mode yang berbeda baik biner maupun BCD. Keistimewaan dari counter ini adalah harga counter dapat dibaca setiap saat tanpa mengganggu kerja counter itu sendiri.

Untuk memindahkan data dari sistem atau sebaliknya, PIT 8253 ini dilengkapi dengan data bus buffer. Buffer ini 3-state, bidirectional dan terdiri dari 8 bit. Data bus buffer ini mempunyai tiga fungsi yaitu :

1. Untuk memprogram mode dari 8253.
2. Untuk mengisi harga count register.
3. Untuk membaca harga count yang sedang dihitung.

Read/Write logic menerima sinyal input dari sistem bus untuk mengendalikan operasi timer ini. Komponen ini dalam keadaan enable atau disable ditentukan oleh  $\overline{CS}$ . Beberapa pin yang dipunyai oleh blok ini adalah :

- a.  $\overline{RD}$  (Read) : Logika '0' (low) pada pin ini digunakan untuk membaca harga counter.
- b.  $\overline{WR}$  (Write) : Logika '0' (low) pada pin ini digunakan

untuk mengisi harga counter.

c.  $A_0$  dan  $A_1$  : Input ini biasanya dihubungkan dengan address bus. Pin-pin ini berfungsi untuk memilih satu dari ketiga counter untuk dioperasikan dan memberi address pada control word register untuk memilih mode.

d.  $\overline{CS}$  : Chip select pada logika '0' (low) menyebabkan device enable, dan sebaliknya logika '1' (high) pada pin ini menyebabkan disable.

Kombinasi kelima sinyal ini menghasilkan tabel operasi seperti yang ditunjukkan pada tabel 2.1.

Tabel 2.1. Operasi PIT 8253<sup>6)</sup>

| $\overline{CS}$ | $\overline{RD}$ | $\overline{WR}$ | $A_1$ | $A_0$ |                      |
|-----------------|-----------------|-----------------|-------|-------|----------------------|
| 0               | 1               | 0               | 0     | 0     | Load Counter No. 0   |
| 0               | 1               | 0               | 0     | 1     | Load Counter No. 1   |
| 0               | 1               | 0               | 1     | 0     | Load Counter No. 2   |
| 0               | 1               | 0               | 1     | 1     | Write Mode Word      |
| 0               | 0               | 1               | 0     | 0     | Read Counter No. 0   |
| 0               | 0               | 1               | 0     | 1     | Read Counter No. 1   |
| 0               | 0               | 1               | 1     | 0     | Read Counter No. 2   |
| 0               | 0               | 1               | 1     | 1     | No-Operation 3-State |
| 1               | x               | x               | x     | x     | Disable 3-State      |
| 0               | 1               | 1               | x     | x     | No-Operation 3-State |

<sup>6)</sup> Ibid, hlm. 232.

Control word register dipilih jika  $A_0A_1 = 11$ , yang kemudian menerima informasi dari data bus buffer dan menyimpan dalam register. Informasi yang disimpan dalam register ini memilih mode tiap counter dan memilih operasi hitung biner atau BCD. Control word register hanya dapat ditulis dan tidak dapat dibaca.

Fungsi lengkap dari PIT 8253 harus diprogram dalam software. Sekumpulan control word harus dikirimkan dari CPU untuk memberi harga awal setiap counter. Control word yang dikirim berisi program untuk memilih mode, urutan program dan pemilihan operasi hitung biner atau BCD. Sekali diprogram PIT 8253 siap untuk beroperasi sesuai dengan yang diinginkan.

Setiap counter dalam PIT 8253 dapat diprogram sendiri-sendiri dengan menuliskan control word ke dalam control word register. Control word yang dikirimkan mempunyai nilai control word format seperti yang ditunjukkan pada gambar 2.6.

| $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ |
|-------|-------|-------|-------|-------|-------|-------|-------|
| SC1   | SC0   | RL1   | RLO   | M2    | M1    | M0    | BCD   |

Gambar 2.6. *Format Control Word*<sup>7)</sup>

<sup>7)</sup> Ibid, hlm. 234.

## a. SC - Select Counter

| SC1 | SC2 |                  |
|-----|-----|------------------|
| 0   | 0   | Select Counter 0 |
| 0   | 1   | Select Counter 1 |
| 1   | 0   | Select Counter 2 |
| 1   | 1   | Illegal          |

## b. RL - Read/Load

| RL1 | RL0 |   |
|-----|-----|---|
| 0   | 0   | Counter Latching operation.   |
| 1   | 0   | Read/Load most significant byte only.                               |
| 0   | 1   | Read/Load least significant byte only.                              |
| 1   | 1   | Read/Load least significant byte first, then most significant byte. |

## c. M - MODE

| M2 | M1 | M0 |        |
|----|----|----|--------|
| 0  | 0  | 0  | Mode 0 |
| 0  | 0  | 1  | Mode 1 |
| x  | 1  | 0  | Mode 2 |
| x  | 1  | 1  | Mode 3 |
| 1  | 0  | 0  | Mode 4 |
| 1  | 0  | 1  | Mode 5 |

## d. BCD

|   |  |
|---|--|
| 0 | Binary Counter 16-bits                         |
| 1 | Binary Coded Decimal (BCD) Counter (4 Decades) |

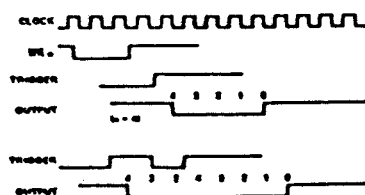
Gambar 2.7. Definisi Control Word Register<sup>8)</sup><sup>8)</sup> Loc.cit.

Penjelasan dari control word register diatas dapat dilihat pada gambar 2.7.

Pada perencanaan alat, mode PIT 8253 yang dipilih adalah mode 1. Definisi dari mode 1 tersebut adalah sebagai berikut :

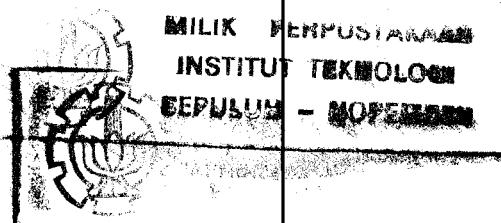
#### MODE 1 (PROGRAMMABLE ONE-SHOT)

Sinyal output akan low ketika masukan GATE mendapat trigger. Dan sinyal output akan menjadi high pada akhir hitungan. Apabila suatu hitungan baru dipindahkan pada saat sinyal output low, hal tersebut tidak akan berpengaruh pada durasi pulsa one-shot sampai trigger berikutnya. Hitungan pada saat itu dapat dibaca tanpa mempengaruhi pulsa one-shot. One-shot dapat ditrigger kembali, karena itu sinyal output akan tetap low untuk menghitung penuh setelah setiap masukan GATE mendapatkan trigger. Untuk lebih jelasnya lihat gambar 2.8.



Gambar 2.8. *Timing Diagram Mode 1 PIT 8253*<sup>9)</sup>

<sup>9)</sup> Ibid, hlm. 236.



## II.4. ANALOG TO DIGITAL CONVERTER

Analog to Digital Converter berfungsi untuk mengubah tegangan analog yang ada pada inputnya menjadi data digital pada outputnya.

Ada berbagai tipe ADC yang masing-masing mempunyai kecepatan konversi, akurasi, stabilitas dan harga yang berbeda-beda. Secara umum dapat dibedakan 2 golongan ADC, yaitu golongan open-loop (tanpa feedback) dan closed-loop (terdapat feedback). Masing-masing mempunyai keunggulan dan kelemahan.

Termasuk dalam golongan open-loop adalah tipe Flash ADC, Slope Converter, Dual Slope Converter. Sedangkan yang termasuk dalam golongan closed-loop adalah Single Counter ADC, Tracking ADC, Successive Approximation ADC.

Mengingat banyaknya tipe ADC, maka dalam pembahasan ini hanya akan menjelaskan Successive Approximation ADC, karena dalam perencanaan peralatan ini digunakan ADC jenis ini.

Successive Approximation ADC merupakan golongan ADC yang closed-loop. Waktu konversi pada ADC ini selalu tetap, tidak bergantung pada besarnya sinyal analog (input). Untuk  $n$  bit ADC tipe ini, diperlukan waktu konversi  $n+1$  siklus. Tetapnya waktu konversi ini juga merupakan salah satu keuntungan penggunaan ADC tipe ini.

Gambar 2.9 menunjukkan contoh ADC tipe successive approximation insd Pada Gambar tersebut diambil contoh 4 bit

successive approximation ADC yang menggunakan 4 bit register dan DAC dengan kenaikan per step sebesar 1 volt.

JK flip-flop A, B, C dan D membentuk register yang akan memberi input digital pada DAC, yang pada akhir proses konversi akan menahan hasil konversi. Sebuah ring counter yang terdiri dari D flip-flop V, W, X, Y dan Z memberikan sequence dengan clock input.

Operasi rangkaian ADC tipe successive approximation dapat dijelaskan sebagai berikut :

Sebuah pulsa start diberikan untuk meng-clear register A, B, C dan D dan men-set control flip-flop M ke kondisi high. Selain itu filp-flop W, X, Y dan Z akan di-clear dan flip-flop V akan di-set sehingga ring counter berada pada kondisi awal 00001. Dengan output register nol, maka tegangan  $V_A$  akan nol, sehingga output comparator adalah high (COMP=1).

Dengan M=1, AND-gate 9 dapat melewatkan pulsa clock ke ring counter. Pulsa clock ini membawa ring counter ke kondisi 10000 dengan Z=1.

Dengan COMP=1 dan Z=1, NAND-gate 1 akan low, yang akan men-set flip-flop D menjadi high. Bit ini merupakan MSB input ke DAC yang menyebabkan tegangan output DAC ( $V_A$ ) menjadi 8 volt. Bila tegangan ini melebihi  $V_A$ , COMP akan low (dan COMP akan high), yang menyebabkan transisi negatif pada output NAND-gate 9 yang akan memberikan pulsa clock flip-flop D kembali ke kondisi low. Sebaliknya jika tegangan 8 volt tidak





kembali ke low. Jika  $V_A'$  tidak melebihi  $V_A$ , maka flip-flop C tetap pada kondisi high.

Pulsa clock ketiga dan keempat mengulangi proses di atas pada flip-flop B dan flip-flop A. Setelah pulsa keempat, register A, B, C dan D berisi nilai digital dari tegangan output analog  $V_A$ .

Pulsa kelima membawa ring counter kembali ke kondisi 00001 dengan flip-flop V=1. Saat V menjadi high, akan dihasilkan transisi negatif pada clock flip-flop M dan menyebabkan flip-flop M menjadi low. Ini membuat AND-gate 9 menjadi tidak aktif sehingga tidak ada lagi pulsa clock yang dapat mempengaruhi ring counter.





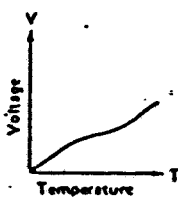
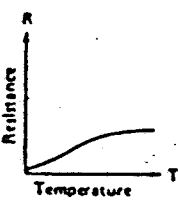
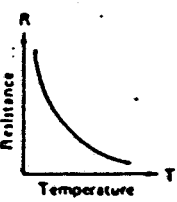
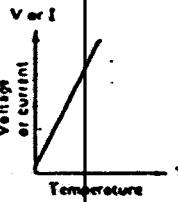
Secara keseluruhan operasi di atas memerlukan 5 periode clock, tetapi waktu konversi ADC sendiri sebenarnya hanya 4 periode clock. Clock kelima digunakan untuk mengakhiri operasi konversi.

## II.5. SENSOR TEMPERATUR

Sensor temperatur dewasa ini ada berbagai macam bentuk diantaranya adalah thermocoupep, resistance temperature detector (RTD), thermistor dan IC sensor. Masing-masing bentuk ini memiliki keunggulan dan kelemahan tersendiri. Pada gambar 2.10 memperlihatkan dan membandingkan kelebihan dan kekurangan masing-masing sensor. Dalam bagian ini hanya akan dibahas sensor temperatur menggunakan IC, karena dalam perencanaan

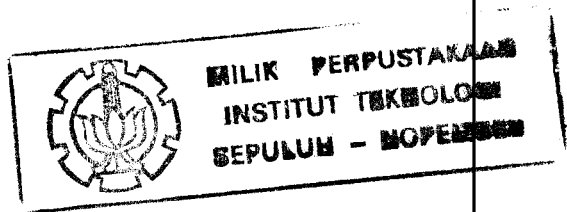
alat, sensor ini yang dipakai. IC sensor temperatur dipilih karena linier, murah harganya, bentuk fisiknya kecil dan sensitivitasnya tinggi.

Pada dasarnya IC sensor temperatur ada dua jenis yaitu sumber tegangan sensitif terhadap temperatur dan sumber arus sensitif terhadap temperatur.

|               | Thermocouple   | RTD   | Thermistor   | IC Sensor   |
|---------------|--|---|--|---|
|               |   |    |   |    |
|               |    |   |    |    |
| Advantages    | <ul style="list-style-type: none"><li><input type="checkbox"/> Self-powered</li><li><input type="checkbox"/> Simple</li><li><input type="checkbox"/> Rugged</li><li><input type="checkbox"/> Inexpensive</li><li><input type="checkbox"/> Wide variety</li><li><input type="checkbox"/> Wide temperature range</li></ul> | <ul style="list-style-type: none"><li><input type="checkbox"/> Most stable</li><li><input type="checkbox"/> Most accurate</li><li><input type="checkbox"/> More linear than thermocouple</li></ul>  | <ul style="list-style-type: none"><li><input type="checkbox"/> High output</li><li><input type="checkbox"/> Fast</li><li><input type="checkbox"/> Two-wire ohms measurement</li></ul>  | <ul style="list-style-type: none"><li><input type="checkbox"/> Most linear</li><li><input type="checkbox"/> Highest output</li><li><input type="checkbox"/> Inexpensive</li></ul>   |
| Disadvantages | <ul style="list-style-type: none"><li><input type="checkbox"/> Non-linear</li><li><input type="checkbox"/> Low voltage</li><li><input type="checkbox"/> Reference required</li><li><input type="checkbox"/> Least stable</li><li><input type="checkbox"/> Least sensitive</li></ul>                                      | <ul style="list-style-type: none"><li><input type="checkbox"/> Expensive</li><li><input type="checkbox"/> Power supply required</li><li><input type="checkbox"/> Small <math>\Delta R</math></li><li><input type="checkbox"/> Low absolute resistance</li><li><input type="checkbox"/> Self-heating</li></ul> | <ul style="list-style-type: none"><li><input type="checkbox"/> Non-linear</li><li><input type="checkbox"/> Limited temperature range</li><li><input type="checkbox"/> Fragile</li><li><input type="checkbox"/> Power supply required</li><li><input type="checkbox"/> Self-heating</li></ul> | <ul style="list-style-type: none"><li><input type="checkbox"/> <math>T &lt; 200^{\circ}\text{C}</math></li><li><input type="checkbox"/> Power supply required</li><li><input type="checkbox"/> Slow</li><li><input type="checkbox"/> Self-heating</li><li><input type="checkbox"/> Limited configurations</li></ul> |

Gambar 2.10. Perbandingan Sensor Temperatur<sup>11)</sup>

11) Charles A. Schuler, William L. McNamee, Industrial Electronics and Robotics, 1-st, McGraw-Hill, Singapore, 1986, hlm. 214.



### II.5.1. SENSOR IC TEMPERATUR

Pada IC sensor sumber tegangan sensitif terhadap temperatur, tegangan output akan berubah terhadap perubahan temperatur. IC sensor yang termasuk jenis ini meliputi LM 34, LM 35 dan LM 135/LM 235/LM 335 buatan National Semiconductor.

Untuk pemilihan sensor yang sesuai dengan spesifikasi yang dibutuhkan dapat dilihat pada tabel 2.2. Dalam tugas akhir ini menggunakan sensor temperatur LM 335 karena komponen ini mudah didapat dan harganya murah.

Rangkaian IC sensor ini mampu mengeliminasi kesalahan linier yang merupakan kekurangan dari sensor thermocouple dan thermistor. Tetapi kelemahan IC sensor ini adalah timbulnya kesalahan akibat self-heating, dan untuk mengurangi efek ini adalah mengoperasikan IC pada arus yang minimum namun cukup untuk mendrive sensor, selain itu tahanan potensiometer dikalibrasi pada temperatur kerja maksimum.

Gambar sensor LM 335 diperlihatkan pada gambar 2.11. IC bekerja sebagai zener dua terminal dengan output sebanding dengan  $10 \text{ mV}/^{\circ}\text{K}$ . Impedansi dinamik kurang dari  $1 \Omega$  dan beroperasi pada jangkauan arus  $400 \mu\text{A}$  sampai  $5\text{mA}$ .

Konfigurasi rangkaian yang digunakan pada tugas akhir ini adalah sensor dengan satu tahanan trimpot kalibrasi titik

Tabel 2.2. Daftar Pemilihan Sensor Temperatur<sup>12)</sup>

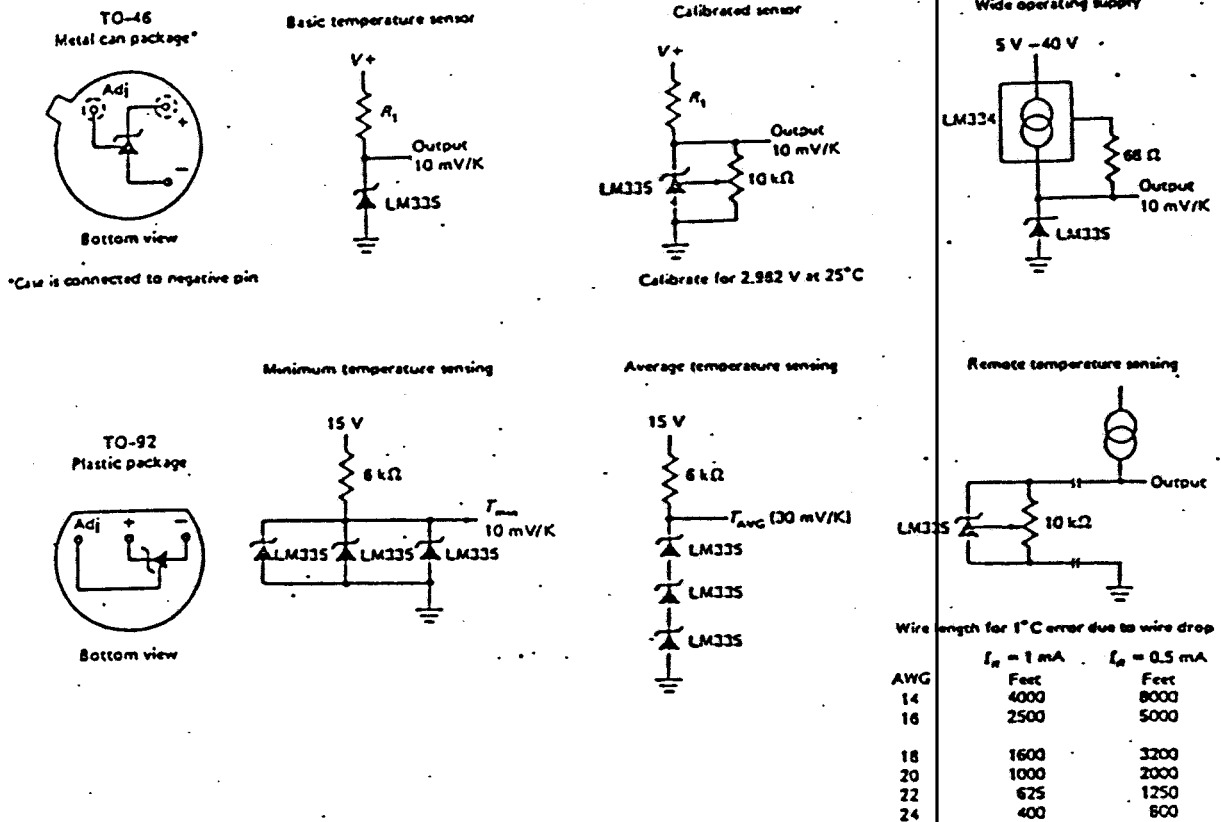
| Part    | Temp. Range     | *Accuracy | Output Scale          |
|---------|-----------------|-----------|-----------------------|
| LM34A   | -50°F to +300°F | ±2.0°F    | 10 mV/°F              |
| LM34    | -50°F to +300°F | ±3.0°F    | 10 mV/°F              |
| LM34CA  | -40°F to +230°F | ±2.0°F    | 10 mV/°F              |
| LM34C   | -40°F to +230°F | ±3.0°F    | 10 mV/°F              |
| LM34D   | +32°F to +212°F | ±4.0°F    | 10 mV/°F              |
| LM35A   | -55°C to +150°C | ±1.0°C    | 10 mV/°C              |
| LM35    | -55°C to +150°C | ±1.5°C    | 10 mV/°C              |
| LM35CA  | -40°C to +110°C | ±1.0°C    | 10 mV/°C              |
| LM35C   | -40°C to +110°C | ±1.5°C    | 10 mV/°C              |
| LM35D   | 0°C to +100°C   | ±2.0°C    | 10 mV/°C              |
| LM134-3 | -55°C to +125°C | ±3.0°C    | I <sub>SET</sub> = °K |
| LM134-6 | -55°C to +125°C | ±6.0°C    | I <sub>SET</sub> = °K |
| LM234-3 | -25°C to +100°C | ±3.0°C    | I <sub>SET</sub> = °K |
| LM234-6 | -25°C to +100°C | ±6.0°C    | I <sub>SET</sub> = °K |
| LM135A  | -55°C to +150°C | ±1.5°C    | 10 mV/°K              |
| LM135   | -55°C to +150°C | ±2.0°C    | 10 mV/°K              |
| LM235A  | -40°C to +125°C | ±1.5°C    | 10 mV/°K              |
| LM235   | -40°C to +125°C | ±2.0°C    | 10 mV/°K              |
| LM335A  | -40°C to +100°C | ±2.0°C    | 10 mV/°K              |
| LM335   | -40°C to +100°C | ±4.0°C    | 10 mV/°K              |
| LM3911  | -25°C to +85°C  | ±10.0°C   | 10 mV/°K (or °F)      |

\*Note: Accuracy is measured over T(Min) to T(Max) uncalibrated

Note: The LM134/234/334 3-Terminal Adjustable current sources Datasheet can be found in Linear 1, Section 1.

tunggal (single point), karena output sensor sebanding dengan penunjukkan temperatur absolut yaitu 2,7315 volt pada 273,15 °K = 0 °C. Penyimpangan tegangan output terhadap temperatur hanyalah simpangan slope (gradien), karena IC sensor ini linier maka sekali kalibrasi pada salah satu temperatur berlaku untuk semua temperatur.

<sup>12)</sup> ...., Linear Databook 2, National Semiconductor Corp., Santa Clara, California, USA, hlm. 6-3.

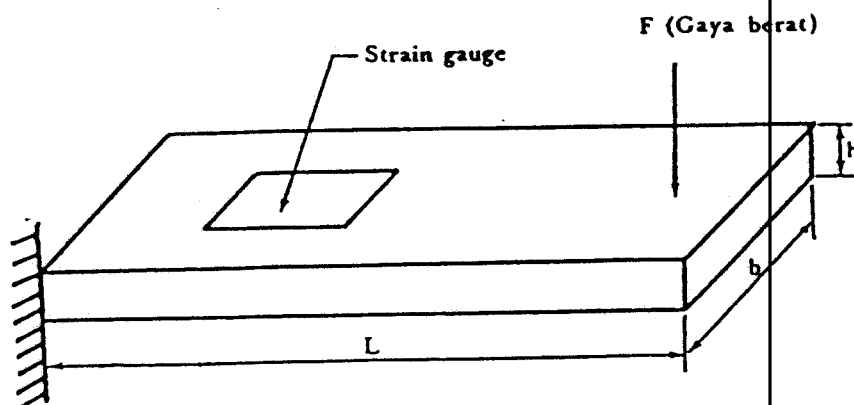


Gambar 2.11. IC Sensor Temperatur LM 335<sup>13)</sup>

## II.6. SENSOR GAYA BERAT

Untuk mengukur regangan yang terjadi pada specimen akibat gaya berat digunakan strain gauge. Gambar 2.12 memperlihatkan cara mengukur gaya berat dengan efek bending. Pada pengukuran gaya berat ini strain gauge diletakan pada ujung specimen yang diklam. Dengan mengetahui regangan

<sup>13)</sup> Charles A. Schuler, William L. McNamee, op.cit, hlm. 217.



Gambar 2.12. Pengukuran Gaya Berat

(strain) specimen. kemudian akan didapat besarnya tegangan (stress). Dengan menggunakan hukum Hooke akan didapat hubungan sebagai berikut :<sup>14)</sup>

$$\varepsilon = \frac{\sigma}{E} \quad (2.1)$$

$$\sigma = \frac{M}{W} \quad (2.2)$$

$$M = F \cdot L \quad (2.3)$$

$$W = \frac{b \cdot h^2}{6} \quad (2.4)$$

<sup>14)</sup> Hoffmann, Fundamentals of Strain Gauge Techniques. Hottinger Baldwin Messtechnik, 1974. hlm. 32.

$$\varepsilon = \frac{6.F.L}{b.h^2.E} \quad (2.5)$$

dimana :

$\sigma$  = Tegangan (Stress) yang terjadi [ $N/mm^2$ ]

E = Modulus Elastisitas specimen [ $N/mm^2$ ]

$\varepsilon$  = Regangan (Strain) yang diukur [mm/mm]

M = Momen Bending [N.m]

L = panjang specimen [mm]

W = Modulus Section [ $mm^3$ ]

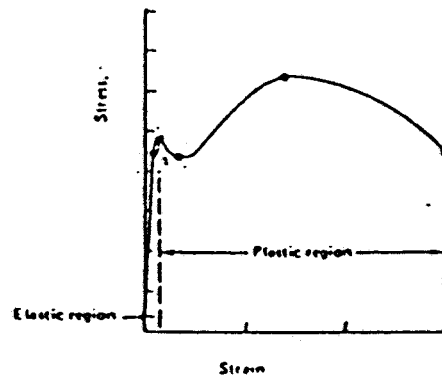
h = Tebal specimen [mm]

b = Lebar specimen [mm].

Persamaan-persamaan diatas hanya berlaku jika specimen masih dalam daerah elastisitas bahan yaitu selama specimen masih mampu kembali ke bentuk semula jika specimen test mengalami regangan. Ketika specimen mengalami regangan sampai masuk daerah plastis yaitu specimen tidak mampu kembali ke bentuk semula sehingga perhitungan dengan persamaan-persamaan diatas tidak berlaku.

Pada perencanaan alat ini specimen dijaga tidak melewati daerah elastisitas bahan. Untuk lebih jelasnya tentang daerah elastis dan plastis suatu bahan ditunjukkan pada gambar 2.13.





Gambar 2.13. *Kurva Stress dan Strain*<sup>15)</sup>

#### II.6.1. KONSEP STRAIN GAUGE

Ada beberapa macam strain gauge yang kita kenal antara lain :

- mechanical strain gauge
- optical strain gauge
- acoustical strain gauge
- electrical strain gauge.

Dalam tugas akhir ini digunakan jenis electrical strain gauge tipe electrical resistance strain gauge. Alat pengukur regangan ini mempunyai kelebihan dalam hal keakuratan hasil pengukuran.

---

<sup>15)</sup> Ibid, hlm. 10.

## II.6.2. PRINSIP KERJA STRAIN GAUGE

Strain gauge bekerja berdasarkan prinsip sifat tahanan listrik suatu kawat penghantar yang akan mengalami perubahan besar tahanan jika dimensi kawat penghantar berubah. Untuk mengukur regangan yang terjadi pada specimen, strain gauge ditempelkan dengan lem perekat khusus pada specimen. Jika specimen mengalami regangan, maka strain gauge juga akan mengalami perubahan tahanan listrik. Jadi strain gauge pada prinsipnya merubah regangan mekanis menjadi perubahan listrik.

Hukum Ohm menyatakan bahwa tahanan listrik suatu kawat penghantar adalah sebanding dengan panjang kawat penghantar dan berbanding terbalik dengan luas penampang penghantar serta bergantung pada tahanan jenis kawat penghantar.

$$R = \rho \frac{L}{A} \quad (2.6)$$

dimana :

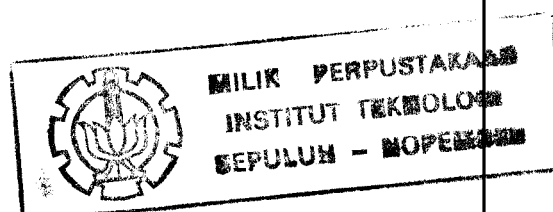
$R$  = tahanan listrik kawat penghantar [ohm]

$L$  = panjang kawat penghantar [mm]

$A$  = Luas penampang kawat penghantar [mm<sup>2</sup>]

$\rho$  = tahanan jenis kawat penghantar [Ohm.mm].

Perubahan tahanan listrik pada suatu strain gauge kemudian dengan perantara rangkaian elektronika, yaitu rangkaian jembatan wheatstone akan dirubah menjadi perubahan tegangan listrik yang sesuai dengan regangan yang terjadi.



### II.6.3. SENSITIVITAS REGANGAN

Tingkat sensitivitas regangan strain gauge, pada dasarnya dipengaruhi oleh dua faktor yakni :

- perubahan dimensi kawat penghantar
- perubahan tahanan spesifik pada specimen alloy kawat penghantar.

Tingkat sensitivitas regangan strain gauge dapat ditentukan berdasarkan persamaan 2.6 yaitu :

$$R = \rho \frac{L}{A}$$

Dengan pendeferensialan dan pembagian terhadap persamaan 2.6 diperoleh :

$$\frac{dR}{R} = \frac{d\rho}{\rho} + \frac{dL}{L} - \frac{dA}{A} \quad (2.7)$$

Bila diketahui regangan transversal yang terjadi akibat peregangannya adalah :

$$\epsilon_t = -\nu \left[ \frac{dL}{L} \right] \quad (2.8)$$

dan diameter kawat setelah peregangannya menjadi :

$$d_f = d_o \left[ 1 - \nu \frac{dL}{L} \right] \quad (2.9)$$

luasan kawat penghantar sebelum dan sesudah peregangannya diketahui :

$$A_o = \frac{\pi}{4} (d_o)^2 \quad (2.10)$$

$$A_f = \frac{\pi}{4} (d_f)^2 \quad (2.11)$$

dimana :

$d_f$  = diameter kawat setelah peregangan [mm]

$d_o$  = diameter kawat sebelum peregangan [mm]

maka perubahan luas penampang kawat penghantar didapat :

$$\frac{dA}{A} = \frac{A_f - A_o}{A_f} = -2 \nu \left[ \frac{dL}{L} \right]^2 \quad (2.12)$$

Substitusi persamaan 2.12 pada persamaan 2.7 didapat :

$$\frac{dR}{R} = \frac{d\rho}{\rho} + \frac{dL}{L} (1 + 2\nu) \quad (2.13)$$

diketahui  $dL/L = \epsilon$ , maka dapat dituliskan sebagai

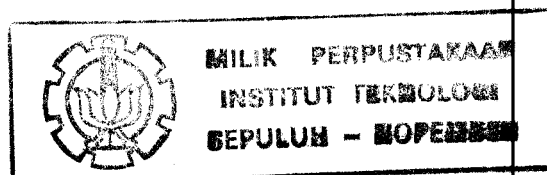
$$S_A = \frac{dR/R}{\epsilon} = (1 + 2\nu) + \frac{d\rho/\rho}{\epsilon} \quad (2.14)$$

dimana :

$S_A$  = sensitivitas regangan material alloy kawat penghantar

$(1+2\nu)$  = perubahan tahanan akibat dimensi kawat penghantar

$d\rho/\rho$  = perubahan tahanan akibat perubahan tahanan spesifik specimen alloy kawat penghantar.



#### II.6.4. FAKTOR GAUGE

Bila berdasarkan persamaan 2.14 dinyatakan bahwa  $S_A$  adalah sensitivitas gauge yang pada dasarnya dibentuk dari sejumlah kawat penghantar, dapat atau sering dikatakan bahwa sensitivitas specimen alloy kawat penghantar adalah sama dengan faktor gauge. Dimana faktor gauge didefinisikan sebagai perubahan tahanan listrik per unit tahanan awalnya dibagi dengan regangan yang terjadi, atau dalam perumusan dinyatakan seperti persamaan berikut :

$$S_g = \frac{\Delta R/R}{\epsilon} \quad (2.15)$$

dimana  $\epsilon$  adalah regangan aksial.

Dengan kata lain faktor gauge adalah sensitivitas perubahan tahanan  $R$  terhadap perubahan regangan, dimana makin besar harga sensitivitas regangan alloy kawat penghantar atau faktor gauge maka pengukuran akan makin akurat dan strain gauge menjadi sensitif.

#### II.6.5. PENGARUH TEMPERATUR

Sering pengukuran dilakukan pada tempat dimana temperatur kerja berubah. Perubahan temperatur akan mengakibatkan :

1. Sensitivitas regangan ( $S_A$ ) dari material strain gauge berubah.
2. Panjang strain gauge berubah memanjang/mengkerut

$$(\Delta L/L = \alpha \Delta T).$$

3. Material (specimen) yang diukur memanjang/mengkerut

$$(\Delta L/L = \beta \Delta T).$$

4. Tahanan dari strain gauge berubah ( $\Delta R/R = \gamma \Delta T$ ).

Pengaruh dari perubahan temperatur terhadap perubahan tahanan dengan memperhatikan hal diatas adalah :

$$\left( \frac{\Delta R}{R} \right)_{\Delta T} = (\beta - \alpha) S_g \Delta T + \gamma \Delta T \quad (2.16)$$

dimana :

$\alpha$  = koefesien regangan panas gauge material

$\beta$  = koefesien regangan panas material (specimen)

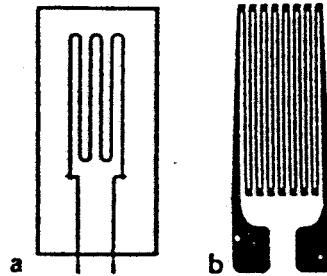
$\gamma$  = koefesien resistivitas gauge

$S_g$  = faktor gauge.

#### II.6.6. SUSUNAN STRAIN GAUGE

Penggunaan strain gauge yang cukup luas menghendaki adanya berbagai bentuk dan susunan strain gauge didasarkan pada kondisi tegangan yang diselidiki atau diteliti pada specimen dan jenis pembebanan yang diberikan.

Untuk kondisi regangan satu arah yang searah dengan arah utama digunakan strain gauge tipe satu elemen. Sedangkan untuk kondisi regangan dua atau tiga dimensi digunakan strain



Gambar 2.14. *Strain Gauge satu sumbu.*<sup>16)</sup>

(a) *Kawat.*

(b) *Foil.*

gauge tipe rosette. Pada tugas akhir ini digunakan strain gauge tipe satu elemen (sumbu) seperti yang terlihat pada gambar 2.14.

#### II.6.7. PEMASANGAN STRAIN GAUGE PADA JEMBATAN WHEATSTONE

Pada pengukuran regangan ini digunakan jembatan wheatstone, dimana terdapat satu lengan aktif dan penerapan teknik kompensasi suhu dengan memasang dummy gauge. Maksud dari kompensasi suhu pada jembatan wheatstone adalah untuk mengeliminir perubahan temperatur yang terjadi pada gauge aktif. Pemasangan strain gauge pada jembatan wheatstone seperti gambar 2.15.

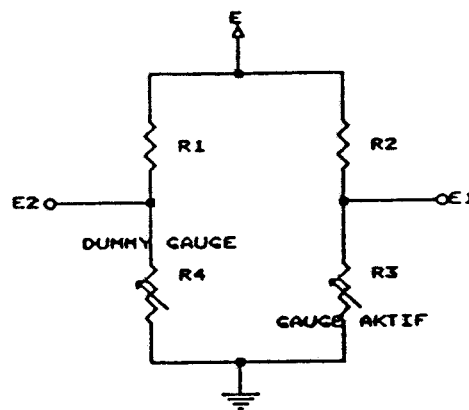
<sup>16)</sup> William D. Cooper, *Instrumentasi Elektronik dan teknik Pengukuran*, Edisi ke-2, Penerbit Erlangga, 1985, hlm. 392.

Dummy gauge yang diletakkan pada salah satu lengan jembatan adalah bersifat non aktif atau dengan kata lain dummy gauge hanya akan aktif terhadap pengaruh temperatur sekeliling.

Pengukuran regangan yang mengakibatkan perubahan resistansi pada gauge aktif sebesar  $\Delta R$  dan memberikan perbedaan tegangan sebesar (misalkan  $R_1 = R_2 = R_3 = R$ )<sup>17)</sup> :

$$E_1 - E_2 = E \frac{\Delta R}{4R} \quad (2.17)$$

Persamaan diatas memperlihatkan bahwa  $E$  harus dibuat cukup besar untuk memaksimumkan tegangan keluaran diferensial jembatan  $E_1 - E_2$ .



Gambar 2.15. Pemasangan Strain Gauge<sup>18)</sup>

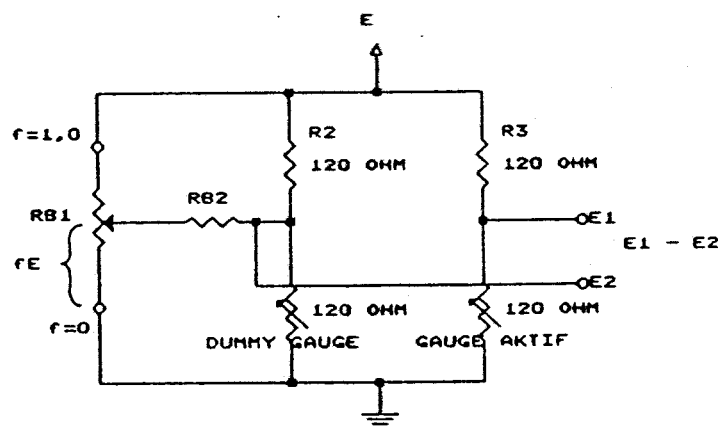
17) Robert F. Coughlin, Frederick F. Driscoll, Penguat Operasional dan Rangkaian Terpadu, Penerbit Erlangga, 1985, hlm. 181.

18) Hoffmann, op.cit, hlm. 18.



### II.6.8. MENYEIMBANGKAN SEBUAH JEMBATAN WHEATSTONE

Pada Gambar 2.16 jika resistansi strain gauge  $120\ \Omega$  yang berfungsi sebagai pengukur regangan dan kompensasi suhu. Untuk melengkapi jembatan tersebut kita memasang tahanan  $120\ \Omega$ , 1 % sehingga range tahanan tersebut  $118,8\ \Omega$  sampai  $121,200\ \Omega$ . Keduanya harus disamakan untuk menyeimbangkan jembatan wheatstone tersebut. Untuk melakukannya kita memasang sebuah tahanan variabel dan sebuah tahanan. Secara teoritis, tahanan tersebut kita atur sedemikian sehingga jembatan dalam keadaan setimbang.



Gambar 2.16. Jembatan dengan Rangkaian Penyeimbang<sup>19)</sup>

<sup>19)</sup> Ibid, hlm. 184.

Untuk memeriksa cara kerja jembatan dalam gambar 2.16, misalkan bahwa tahanan lengan jembatan  $R_2$  dan  $R_3$  sama besar, sampai ketelitian meliputi  $\pm 1\%$ . Resistansi pengukuran regangan tersebut harus mempunyai resistansi yang sama meliputi beberapa miliohm jika pengukuran yang bekerja tidak mengalami regangan.

Tahanan  $R_{B1}$  adalah pot linier 3/4 - lilitan yang biasa. Resistansinya harus sekitar 1/10 atau lebih kecil dari  $R_{B2}$  sedemikian rupa sehingga tegangan  $fE$  hanya tergantung pada  $E$  dan pecahan desimal  $f$  saja. Hanya  $f$  yang berubah-ubah dari 0 sampai 1,0 pada saat pot disetel dari ujung yang satu ke ujung yang lainnya.

Tahanan  $R_{B2}$  harus dipilih lebih besar 10 kali masing-masing pengukuran atau lebih, dan juga 10 kali  $R_{B1}$  atau lebih. Dalam keadaan ini  $R_{B2}$  tidak membebani pekerjaan  $R_{B1}$  sebagai pembagi tegangan. Lagipula, ukuran  $R_{B2}$  menentukan arus penyeimbangan maksimum yang dapat dialiri ke dalam, atau diambil ke luar dari simpul  $E_2$ .  $f$  penyetel pot menentukan berapa banyak arus maksimum yang dialirkan ke dalam atau diambil ke luar.

Kerja penyeimbangan diringkas dengan mengamati bahwa jika  $f > 0,5$ , suatu arus kecil dimasukkan ke dalam simpul  $E_2$  dan mengalir melalui pengukur suhu ke ground, yang membuat  $E_2$  menjadi lebih positif. Jika  $f < 0,5$  arus tersebut diambil dari simpul  $E_2$ , yang memperbesar arus melalui  $R_2$ , membuat  $E_2$  menjadi lebih positif. Dalam sebuah jembatan sesungguhnya, mulai-

lah  $R_{B2}$  dengan  $10\text{ K}\Omega$  dan  $R_{B1}$  dengan  $1\text{ K}\Omega$ . Jika perubahan  $V_o$  lebih besar dari yang diinginkan, perbesar  $R_{B2}$  sampai  $100\text{ K}\Omega$  dan periksa kembali tindakan penyeimbangannya. Harga akhir dari  $R_{B2}$  dipilih berdasarkan percobaan tergantung pada derajat tidak-seimbang antara  $R_2$  dan  $R_3$ .



### BAB III

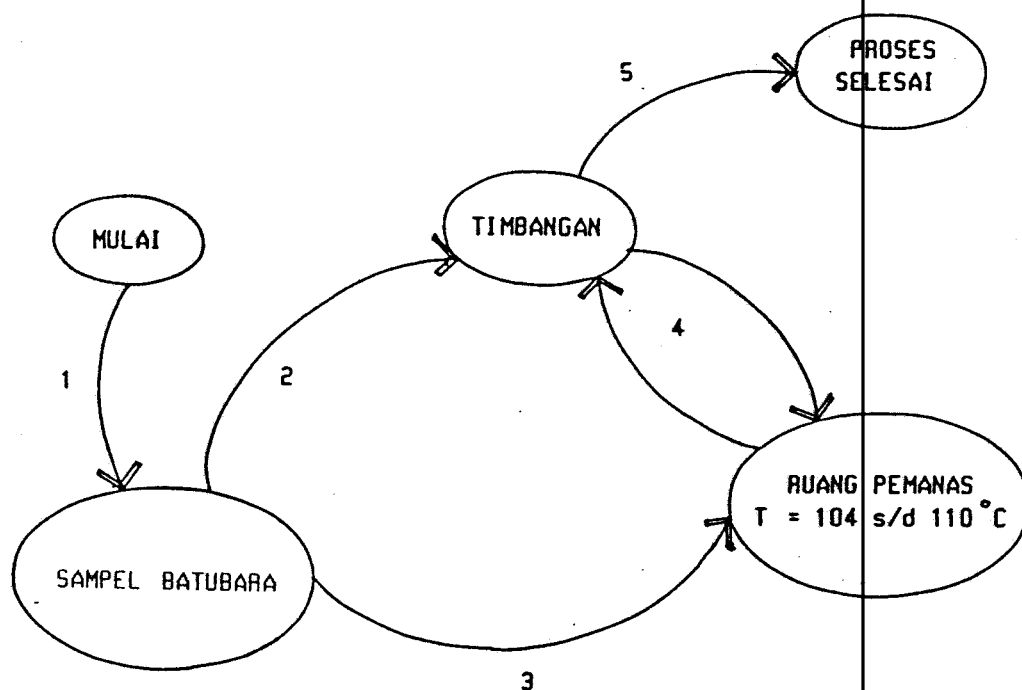
## PERENCANAAN SISTEM PENENTUAN KANDUNGAN AIR DALAM BATUBARA

---

Menentukan kandungan air dalam batubara sangat penting karena mempengaruhi performance bahan bakar tersebut terhadap proses pembakaran dan proses konversi batubara menjadi bahan bakar gas atau bahan bakar cair. Hal-hal yang perlu diperhatikan pada sistem penentuan kandungan air dalam batubara adalah sebagai berikut :

1. Ruang pemanas.
2. Pemanas.
3. Timbangan.

Gambar 3.1 menunjukkan diagram alir dari perencanaan sistem penentuan kandungan air dalam batubara. Mula-mula diambil sampel batubara (maksimum 20 gram), kemudian ditimbang. Setelah diketahui berat sebenarnya sampel batubara dipanaskan pada temperatur kerja yaitu berkisar antara 104 °C sampai dengan 110 °C. Kemudian selama dipanaskan pada temperatur kerja ini sampel batubara ditimbang sampai diperoleh pengukuran berat yang konstan. Selang waktu penimbang saat dipanaskan pada temperatur kerja dapat di-set.



Gambar 3.1. Diagram Alir Sistem Penentuan Kandungan Air dalam Batubara

Yang perlu diperhatikan untuk memperoleh hasil penentuan kandungan air yang teliti dan akurat adalah sebagai berikut :

1. Alat timbangan harus selalu dalam keadaan standar, setiap saat perlu kalibrasi/disesuaikan dengan batu timbangan yang telah diketahui beratnya.
2. Temperatur kerja dari ruang pemanas di-set antara 104 °C sampai dengan 110 °C. Apabila temperatur operasi diatas 110 °C akan terjadi penguapan volatile matter (zat yang mudah menguap) sehingga

menghasilkan pengukuran yang lebih besar dari sebenarnya. Apabila temperatur operasi dibawah 104 °C maka tidak semua air dalam batubara habis menguap, sehingga hasilnya menjadi lebih kecil.

3. Pendekatan paling baik diperoleh bila jumlah Sampel yang diambil dari Batubara sangat banyak. Secara teoritis sampel pengukuran yang banyaknya tak berhingga akan memberikan hasil yang lebih baik.

Kandungan air dalam batubara dinyatakan dalam persen berat yaitu :

$$M = \frac{B_0 - B_1}{B_0} \times 100 \%$$

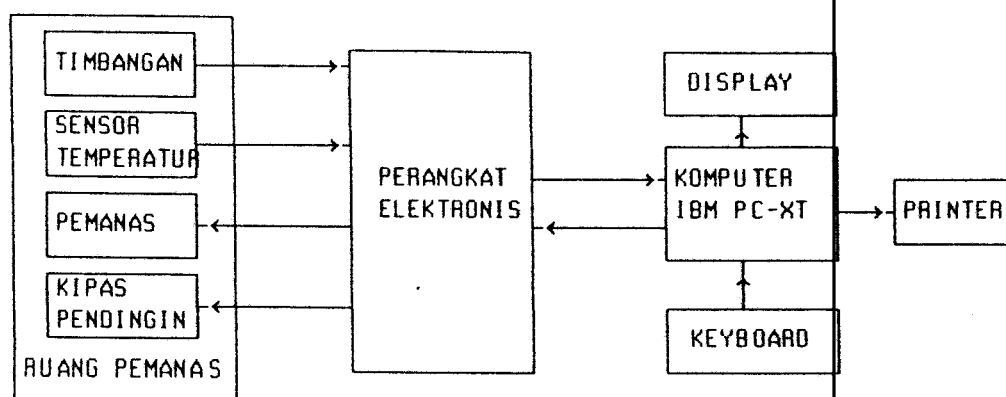
dimana :

M = Kandungan air dalam batubara [%]

B<sub>0</sub> = Berat awal pada temperatur kamar 29,6 °C [gram]

B<sub>1</sub> = Berat akhir pada temperatur berkisar 104 °C sampai dengan 110 °C [gram].

Gambar 3.2 memperlihatkan blok diagram sederhana dari peralatan yang dirancang dengan bantuan mikrokomputer IBM PC-XT sebagai pengolah data dan pengontrol proses yang berlangsung. Dalam ruang pemanas terdapat beberapa bagian yaitu : timbangan, sensor temperatur, pemanas dan kipas pendingin yang semuanya dikontrol oleh mikrokomputer IBM PC-XT melalui



Gambar 3.2. Blok Diagram dari Peralatan yang Dirancang Berbasis pada Mikrokomputer IBM PC-XT

perangkat elektronis. Bagian dari perangkat elektronis pada perencanaan sistem ini dijelaskan pada bab IV.

Mikrokomputer memonitor besarnya temperatur ruang pemanas dan berat dari sampel batubara dan mengontrol pemanas dalam ruang pemanas untuk memperoleh temperatur kerja. Ruang pemanas yang dipakai pada sistem ini mempunyai volume sesuai dengan kebutuhan dan terisolasi dengan lingkungan sekitarnya untuk menjaga temperatur kerja pada ruang pemanas. Ruang pemanas yang dipakai pada tugas akhir ini adalah oven.

Pemanas dalam ruang pemanas berfungsi untuk menaikkan temperatur ruang pemanas sampai diperoleh temperatur kerja berkisar antara 104 °C sampai 110 °C dengan penyalaan dilakukan oleh PIT 8253. Penyalaan pemanas ini menggunakan

kontrol proporsional dengan sinyal penggeraknya merupakan selisih antara setting temperatur kerja dengan temperatur sesaat yang terdeteksi melalui sensor temperatur LM 335.

Untuk mengukur berat dari sampel batubara dipakai timbangan. Alat timbangan ini memakai sensor strain gauge yang dilekatkan pada ujung specimen yang diklam. Prinsip kerja dari timbangan ini adalah mengukur regangan yang terjadi akibat bending bila pada ujung pada specimen yang tidak diklam bekerja gaya berat. Perubahan regangan sebanding terhadap perubahan berat sesuai dengan kurva stress dan strain pada teori penunjang dengan catatan specimen masih dalam daerah elastisitas bahan. Specimen yang dipakai pada tugas akhir ini dirancang untuk pengukuran berat dalam orde miligram.

Penyaklaran kipas pendingin dilakukan oleh port output PPI 8255 yang dikontrol melalui komputer. Kipas pendingin pada ruang pemanas ini berfungsi untuk menurunkan temperatur ruang pemanas sampai temperatur kamar. Dengan kipas pendingin penurunan temperatur relatif lebih cepat.

Hasil dari penentuan kandungan air dari sejumlah sampel batubara oleh mikrokomputer IBM PC-XT dapat disimpan pada diskette atau dicetak pada printer.



## BAB IV

### PERENCANAAN PERANGKAT ELEKTRONIS

---

Dalam bab perencanaan ini akan dibahas tentang tahap-tahap perencanaan perangkat elektronis dari peralatan yang akan dibuat. Karena teori penunjang dari peralatan dan perencanaan sistem sudah dijelaskan pada bab sebelumnya, maka pada bab perencanaan ini akan langsung diuraikan tentang perencanaan hardware dan software.

#### IV.1. PERENCANAAN HARDWARE

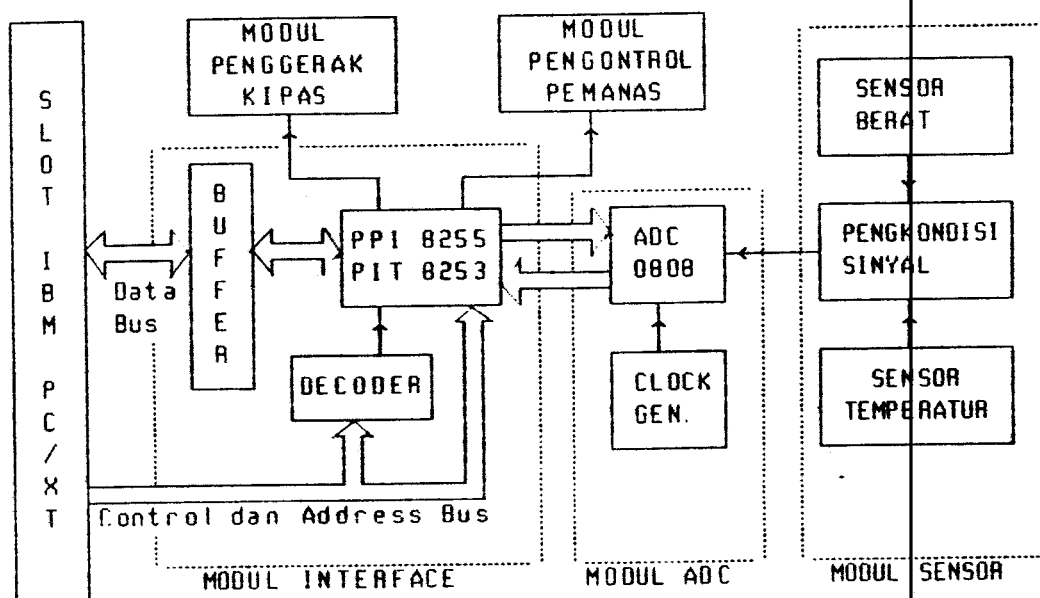
Peralatan yang akan dibuat adalah berfungsi untuk menentukan kandungan air dalam batubara dengan bantuan mikrokomputer IBM PC-XT, parameter-parameter yang dikontrol dalam penentuan kandungan air dalam batubara adalah temperatur dan berat. Peralatan ini bekerja sepenuhnya dengan bantuan mikrokomputer IBM PC-XT sebagai pengolah data input temperatur dan berat yang terbaca dan memberikan respon umpan balik yang harus dilakukan oleh perangkat luar.

Peralatan yang direncanakan dibagi ke dalam modul-modul rangkaian yang mempunyai fungsi dan tugasnya masing-masing sehingga secara keseluruhan merupakan suatu sistem pengukuran kandungan air dalam batubara. Bagian-bagian tersebut adalah :

- Modul interface

- Modul pengontrol pemanas
- Modul penggerak kipas pendingin
- Modul sensor
- Modul Analog to Digital Converter (ADC).

Gambar 4.1 menunjukkan blok diagram perangkat elektronis dari peralatan yang direncanakan yang dikendalikan oleh mikrokomputer IBM PC-XT. Modul interface terdiri dari buffer, decoder, PPI 8255 dan PIT 8253. Data-data digital hasil keluaran modul ADC diolah pada mikrokomputer IBM PC-XT melalui



Gambar 4.1. Blok Diagram Perangkat Elektronis dari Peralatan yang direncanakan

modul interface ini. Dan juga melalui modul interface ini dilakukan pengontrolan pemanas dan penggerak kipas pendingin oleh mikrokomputer IBM PC-XT.

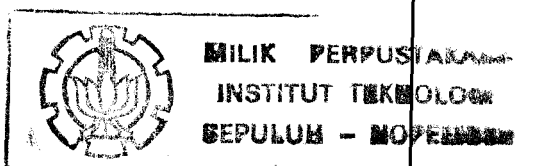
Untuk mengontrol pemanas pada temperatur yang kita inginkan dikontrol melalui modul pengontrol pemanas. Pada modul pengontrolan pemanas ini penyalaan pemanas dikontrol oleh PIT 8253. Kipas pendingin pada Modul penggerak kipas pendingin dapat diaktifkan oleh mikrokomputer IBM PC-XT melalui PPI 8255.

Modul sensor terdiri dari sensor temperatur, sensor berat dan rangkaian pengkondisi sinyal. Sensor temperatur ini digunakan untuk mendeteksi temperatur ruang pemanas. Sensor berat akan merubah gaya berat yang bekerja pada specimen menjadi sinyal listrik. Kemudian Sinyal listrik yang dihasilkan oleh kedua sensor tersebut dimasukkan ke rangkaian pengkodisi sinyal.

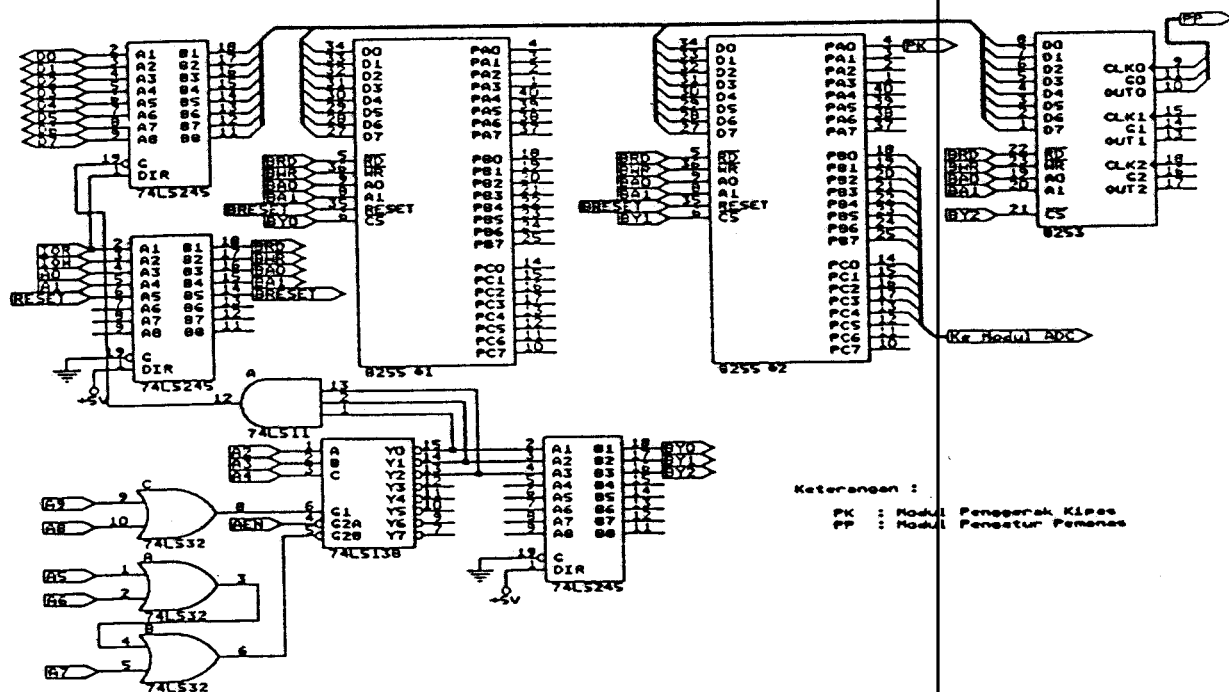
Tegangan keluaran dari modul sensor dimasukkan ke modul ADC untuk dirubah menjadi data-data digital. Modul ADC ini terdiri dari ADC 0808, rangkaian clock dan rangkaian tegangan referensi.

#### IV.1.1. MODUL INTERFACE

Dalam modul ini terdapat rangkaian buffer, rangkaian decoder I/O port, rangkaian PPI 8255 dan rangkaian PIT 8253. Gambar 4.2 menunjukkan rangkaian lengkap dari modul interface.



Modul ini mempergunakan beberapa IC untuk mendukung fungsi dari masing-masing rangkaian. Jenis IC yang dipergunakan disini merupakan IC TTL dengan beberapa seri, yaitu 74LS245 sebagai buffer bidirectional untuk data bus, 74LS11, 74LS32 dan 74LS138 sebagai decoder I/O port PPI 8255 dan PIT 8253. PPI 8255 dihubungkan dengan penggerak kipas penghisap dan Analog to Digital Converter (ADC 0808). Sedangkan PIT 8253 dihubungkan dengan pengatur pemanas.

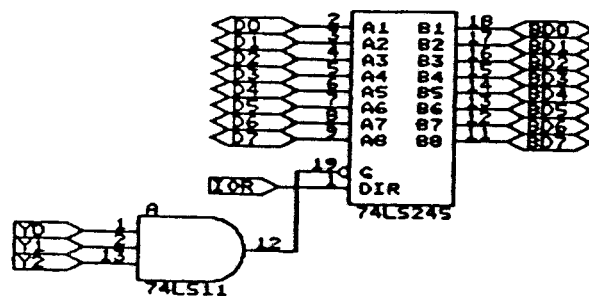


Gambar 4.2. Modul Interface

### RANGKAIAN BUFFER DATA BUS

Rangkaian buffer ini berguna untuk memperkuat sinyal data dari mikrokomputer dan juga untuk mengisolasi sistem data bus dalam mikrokomputer terhadap peralatan luar pada saat peralatan luar tidak diaktifkan.

Sistem data bus mikrokomputer dihubungkan pada sisi A sedangkan data bus peralatan pada sisi B dari buffer bidirectional 74LS245. Chip enable ( $\overline{G}$ ) pada pin 19 IC 74LS245 dihubungkan dengan sinyal 'interface select'. Sinyal interface select ini aktif low dan akan aktif bila mikrokomputer sedang berkomunikasi dengan PPI 8255 dan PIT 8253 atau salah satu dari kedua piranti tersebut. Hal ini dimungkinkan karena adanya IC 74LS11 yang merupakan IC AND gate dengan tiga masukan.



Gambar 4.3. Rangkaian Buffer Data Bus

Pin direction (DIR) IC 74LS245 dihubungkan dengan sinyal  $\overline{\text{IOR}}$ . Sinyal  $\overline{\text{IOR}}$  ini berasal dari sinyal kontrol mikrokomputer IOR (I/O read). Pada saat interface select dan IOR bernilai '0', maka data bus peralatan akan didrive oleh 74LS245 ke sistem data bus mikrokomputer (saat terjadi proses I/O port read cycle) dan sebaliknya bila interface select bernilai '0' dan IOR bernilai '1' maka sistem data bus mikrokomputer akan didrive oleh 74LS245 ke data bus peralatan (saat terjadi proses I/O port write cycle). Gambar 4.3 memperlihatkan rangkaian buffer IC 74LS245.

#### RANGKAIAN DECODER

Dengan memperhatikan lokasi-lokasi I/O port diluar sistem board IBM PC/XT yang belum digunakan, maka dipilih lokasi 0300H hingga 030BH untuk lokasi I/O port PPI 8255 dan PIT 8253 yang digunakan pada peralatan yang direncanakan ini. Lokasi 0300H hingga 0303H untuk PPI 8255 yang pertama sebagai cadangan (reserved) dan lokasi 0304H hingga 0307H untuk PPI 8255 yang kedua, lokasi 0308H hingga 030BH untuk PIT 8253. Tabel 4.1 menunjukkan tabel kebenaran pengkodean I/O port ke PPI 8255 dan ke PIT 8253.

Untuk mewujudkan rangkaian dari tabel kebenaran tersebut, maka akan digunakan IC 74LS32 (berisi 4 buah OR gate masing-masing 2 masukan), 74LS138 (IC decoder 3 bit) dan 74LS11 (berisi 3 buah AND gate dengan 3 masukan).

Tabel 4.1. Tabel Kebenaran Pengkodean I/O Port

| Hex  | A9 | A8 | A7 | A6 | A5 | A4 | A3 | A2 | A1 | A0 | Hubungan   |
|------|----|----|----|----|----|----|----|----|----|----|------------|
| 0300 | 1  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | PA 8255 #1 |
| 0301 | 1  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | PB 8255 #1 |
| 0302 | 1  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | PC 8255 #1 |
| 0303 | 1  | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | CW 8255 #1 |
| 0304 | 1  | 1  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | 0  | PA 8255 #2 |
| 0305 | 1  | 1  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | 1  | PB 8255 #2 |
| 0306 | 1  | 1  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | 0  | PC 8255 #2 |
| 0307 | 1  | 1  | 0  | 0  | 0  | 0  | 0  | 1  | 1  | 1  | CW 8255 #2 |
| 0308 | 1  | 1  | 0  | 0  | 0  | 0  | 1  | 0  | 0  | 0  | Ct1 8253   |
| 0309 | 1  | 1  | 0  | 0  | 0  | 0  | 1  | 0  | 0  | 1  | Ct2 8253   |
| 030A | 1  | 1  | 0  | 0  | 0  | 0  | 1  | 0  | 1  | 0  | Ct3 8253   |
| 030B | 1  | 1  | 0  | 0  | 0  | 0  | 1  | 0  | 1  | 1  | CW 8253    |

P :Port

CW:Control Word

Ct:Counter

Bit-bit A5, A6 dan A7 selalu berharga '0' dijumlahkan oleh 74LS32 untuk mendapatkan nilai keluaran '0', lalu dihubungkan ke input G2B. Input G2A mendapat masukan sinyal AEN (Address Enable) dari mikrokomputer. Sinyal AEN aktif high, AEN akan aktif (high) bila sedang melakukan proses DMA oleh sistem board dengan suatu peralatan I/O. Bila tidak sedang melakukan proses DMA, maka sinyal ini akan selalu low, karenanya bila sistem board tidak sedang melakukan proses DMA, maka input G2A akan mendapatkan sinyal masukan low. Bit-bit A9 dan A8 selalu berharga '1' dijumlahkan oleh 74LS32 untuk mendapatkan nilai keluaran '1', kemudian dihubungkan ke input G1.

Masukan G1, G2A dan G2B menentukan output yang akan dipilih. Keluaran operasi logika persamaan 4.1 dari ketiga

masukan tersebut harus high.

$$E = G1.(G2A + G2B)$$

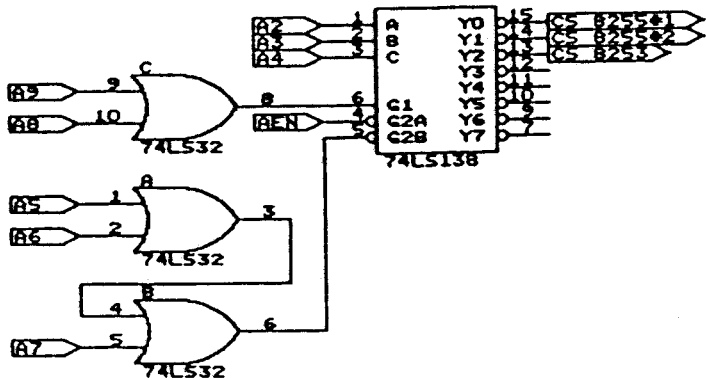
(4.1)

Tabel kebenaran dari IC decoder 74LS138 adalah sebagai berikut :

Tabel 4.2. Tabel Kebenaran 74LS138

| Input |   |   |   | Output  |
|-------|---|---|---|---------|
| E     | C | B | A | Q = Low |
| L     | x | x | x | -       |
| H     | L | L | L | Y0      |
| H     | L | L | H | Y1      |
| H     | L | H | L | Y2      |
| H     | L | H | H | Y3      |
| H     | H | L | L | Y4      |
| H     | H | L | H | Y5      |
| H     | H | H | L | Y6      |
| H     | H | H | H | Y7      |

x: don't care  
H: High  
L: Low



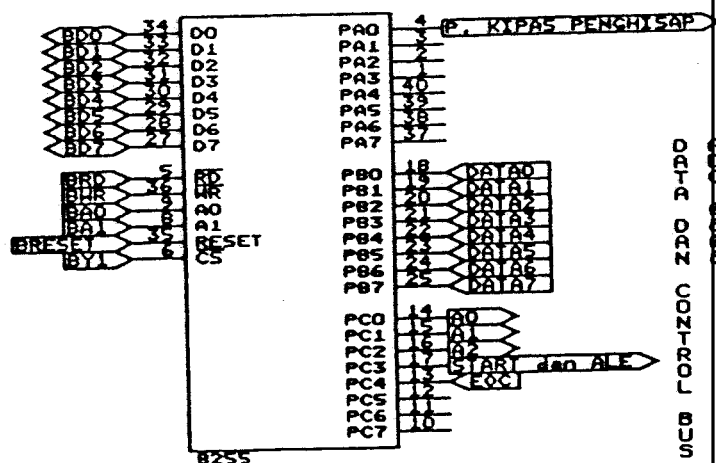
Gambar 4.4. Rangkaian Decoder



Dari tabel kebenaran diatas kemudian diimplementasikan pada rangkaian decoder yang diperlihatkan pada gambar 4.4. Output Y0 digunakan untuk mengaktifkan PPI 8255 yang pertama, output Y1 digunakan untuk mengaktifkan PPI 8255 yang kedua dan output Y2 untuk mengaktifkan PIT 8253.

#### RANGKAIAN PPI 8255

Pada perencanaan ini digunakan dua buah PPI 8255 untuk mendukung fungsi konversi pada modul detektor temperatur dan berat, dan pengaturan relay untuk mengatur pengaktifan kipas penghisap. Gambar 4.5 memperlihatkan rangkaian PPI 8255 yang dioperasikan pada mode 0 yaitu mode input-output.



Gambar 4.5. Rangkaian PPI 8255

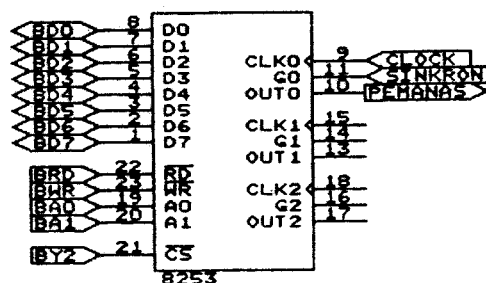
Hubungan port PPI 8255 dengan rangkaian lain ditentukan sebagai berikut :

- PPI 8255 yang pertama sebagai cadangan (reserved).
- PPI 8255 yang kedua
  - Port A, PA0 dihubungkan dengan rangkaian penggerak relay yang akan mengatur pengaktifan kipas penghisap.
  - Port B, PBO-PB7 digunakan sebagai input port untuk data hasil konversi.
  - Port C lower, PC0-PC3 digunakan sebagai output port untuk mengontrol proses konversi pada ADC untuk mendapatkan data pendeteksian temperatur dan gaya berat.
  - Port C upper, PC4 digunakan sebagai input port untuk mengetahui apakah proses konversi ADC telah selesai.

#### RANGKAIAN PIT 8253

Rangkaian PIT 8253 disini dioperasikan pada mode 1 yaitu 'programmable one-shot'. Keluaran OUT dari PIT 8253 dipergunakan untuk mengaktifkan optotriac yang akan mendrive triac. Pengaktifan optotriac ini ditentukan oleh sinyal one-shot yang dihasilkan oleh PIT 8253.

Agar sudut penyalaan triac sesuai dengan phase pada jala-jala listrik maka perlu adanya sinyal sinkronisasi. Sinyal sinkronisasi ini didapatkan dari transformator step-down yang kemudian disearahkan oleh diode dengan konfigurasi full-wave rectifier. Yang diambil adalah tegangan searah positif, yang kemudian diumpankan ke rangkaian detektor



Gambar 4.6. Rangkaian PIT 8253

penyilang-nol pembalik untuk mendapatkan pulsa trigger level TTL. Rangkaian dari PIT 8253 dapat dilihat pada gambar 4.6.

#### IV.1.2. MODUL PENGONTROL PEMANAS

Penyalan pemanas dikontrol oleh PIT 8253 dan penyaklaran dilakukan oleh triac. Untuk mengisolasi antara rangkaian dc dan ac maka dipergunakan optotriac.

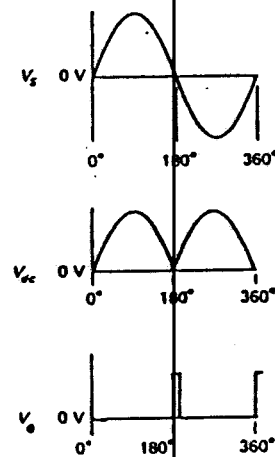
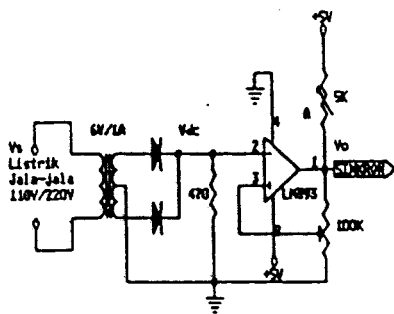
Agar sudut penyalan triac sesuai dengan phase pada jala-jala listrik maka perlu adanya sinyal sinkronisasi. Sinyal sinkronisasi ini didapatkan dari transformator step-down yang kemudian disearahkan oleh diode dengan konfigurasi full-wave rectifier. Yang diambil adalah tegangan searah positif, yang kemudian diumpankan ke rangkaian detektor penyilang-nol pembalik untuk mendapatkan pulsa trigger level TTL.

Rangkaian detektor penyilang-nol pembalik ini disusun

oleh op-amp komparator LM 393. Karena op-amp ini didesain untuk dapat diinterfacekan secara langsung dengan TTL dan CMOS. Selain dari pada itu karena transistor output op-amp ini open collector maka dibutuhkan resistor pull-up pada outputnya.

Pulsa trigger ini digunakan untuk mentrigger pin GATE pada PIT 8253 sebagai titik acuan sudut penyalan. Rangkaian sinkronisasi secara keseluruhan dan sinyal sinkronisasi yang dihasilkannya diperlihatkan pada gambar 4.7.

Sudut penyalan triac ditentukan oleh durasi one-shot yang dihasilkan oleh PIT 8253, awal durasi adalah pada GATE mendapatkan pulsa trigger dari rangkaian sinkronisasi. Selain dari pada itu durasi one-shot juga ditentukan oleh sampling yang dilakukan oleh clock terhadap gelombang sinus jala-jala



Gambar 4.7. Rangkaian dan Sinyal Sinkronisasi

listrik. Pada rangkaian ini, setengah gelombang sinus tersebut dibagi sebanyak OFFH (0-255). Sehingga frekuensi clock yang harus dibangkitkan sebesar :

$$f_{\text{jala-jala}} = 50 \text{ Hz}$$

$$T_{\text{jala-jala}} = \frac{1}{f_{\text{jala-jala}}} = 20 \text{ mdet}$$

$$\frac{T_{\text{jala-jala}}}{2} = 10 \text{ mdet}$$

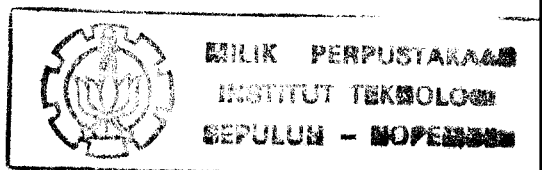
$$T_{\text{clock}} = \frac{10 \text{ mdet}}{256} = 39,1 \mu\text{det}$$

$$f_{\text{clock}} = \frac{1}{T_{\text{clock}}} = 25,58 \text{ KHz}$$

Untuk membuat rangkaian clock dengan frekuensi seperti diatas maka dipilih kapasitor 39nF dan untuk resistornya dapat dicari melalui rumus :

$$f = \frac{1,4}{R C}$$

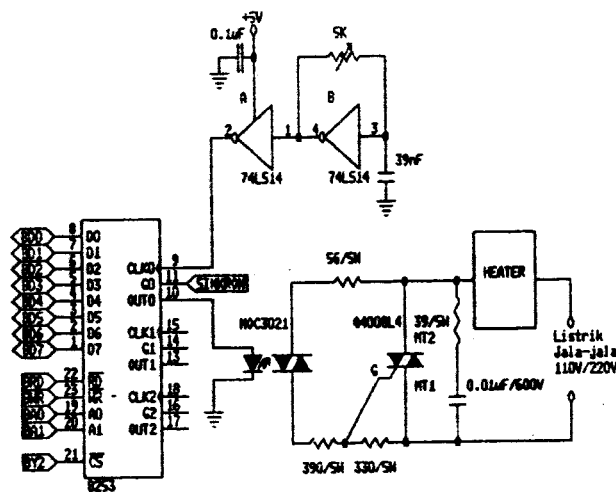
(4.2)



$$\begin{aligned}
 R &= \frac{1,4}{f \cdot C} \\
 &= \frac{1,4}{25,58 \times 10^3 \times 39 \times 10^{-9}} \\
 &= 1403,34 \, \Omega
 \end{aligned}$$

Agar clock tersebut lebih presisi maka resistor yang dipilih adalah resistor variabel jenis multiturn.

Jenis optotriac yang dipergunakan di sini tidak menggunakan rangkaian detektor penyilang-nol didalamnya, yaitu tipe MOC 3021. Penyalaan optotriac ini dilakukan oleh keluaran OUT dari PIT 8253. Untuk mengamankan LED yang ada didalam optotriac maka diberi tahanan seri 470  $\Omega$ , karena LED tersebut memiliki arus trigger maksimum = 15 mA.



Gambar 4.8. PIT 8253 sebagai Rangkaian Pengontrol Pemanas

Siklus penyalaaan optotriac tergantung dari software kontrol yang dipakai. Pada prinsipnya penyalaaan optotriac ini dapat dilakukan setiap saat. Oleh karena itu optotriac yang dipakai tidak memakai rangkaian detektor penyilang-nol.

Optotriac ini aktif apabila diberi sinyal high dari PIT 8253. Pada saat  $n = 0$  maka arus AC dari jala-jala akan dialirkan secara penuh ke beban oleh triac, tetapi pada saat  $n = 255$  maka triac akan memblockng arus AC tersebut.

Setelah optotriac ini aktif, maka akan mengalir arus trigger triac. Arus trigger triac ini juga cukup kecil, maka dari itu antara gate dan MT2 diberi resistor untuk memperkecil arus trigger triac tersebut.

Selain dari pada itu triac tersebut diberi pengaman terhadap beban induktif. Karena pemanas tersebut mempunyai nilai induktif, maka untuk amannya triac tersebut diberi rangkaian 'snubber'. Rangkaian tersebut disusun oleh resistor dan kapasitor seri antara MT1 dan MT2, berfungsi untuk mengurangi pengaruh perubahan tegangan ( $dv/dt$ ) yang melintasi triac ketika aliran arusnya dihambat, hal ini untuk membantu ketika melakukan pemadaman triac. Rangkaian lengkap pengatur pemanas tersebut ditunjukkan pada gambar 4.8.

PEMANAS<sup>20)</sup>

Pemanas yang dipakai pada perencanaan ini memakai sumber energi listrik. Daya listrik yang diperlukan untuk mencapai temperatur  $T_v = 107^\circ\text{C}$  dalam ruang pemanas dari temperatur mula-mula  $T_\infty = 29,6^\circ\text{C}$  dan diameter kawat pemanas (d) 0,02 mm. Asumsi daya listrik yang dibangkitkan dalam kawat dilepas melalui konveksi ke udara adalah :

$$P = I^2 R = q = hA(T_v - T_\infty) \quad (4.3)$$

temperatur film adalah  $T_f = (107 + 29,6)/2 = 68,3^\circ\text{C} \approx 350^\circ\text{K}$ , sehingga diperoleh sifat-sifat udara pada  $T_f = 350^\circ\text{C}$  :

$$\text{koefisien muai volume } (\beta) = 1/T_f = 2,86 \times 10^{-3} \text{ }^\circ\text{K}^{-1}$$

$$\text{viskositas } (\nu) = 20,76 \times 10^{-6} \text{ m}^2/\text{s}$$

$$\text{konduktivitas termal } (k) = 0,03003 \text{ W/m.}^\circ\text{C}$$

$$\text{angka Prandtl } (Pr) = 0,697$$

angka Rayleigh :

$$\begin{aligned} Ra = Gr Pr &= \left[ \frac{g\beta(T_v - T_\infty)d^3}{\nu^2} \right] Pr \\ &= \frac{9,8 \cdot (2,86 \cdot 10^{-3}) \cdot (107 - 29,6) \cdot (0,02 \cdot 10^{-3})^3}{(20,76 \cdot 10^{-6})^2} (0,689) \\ &= 2,807 \times 10^{-5} \end{aligned}$$

<sup>20)</sup> J.P. Holman, E. Jasifi, Perpindahan Kalor, Edisi keenam, Penerbit Erlangga, 1988, hlm. 295.



dari angka Rayleigh, didapat konstanta  $C = 0,675$  dan  $m = 0,058$  sehingga angka Nusselt :

$$\begin{aligned} Nu &= C(Gr Pr)^m \\ &= 0,675.(2,807.10^{-5})^{0,058} \\ &= 0,368 \end{aligned}$$

dan koefisien perpindahan kalor konveksi :

$$\begin{aligned} h &= Nu \left[ \frac{k}{d} \right] \\ &= \frac{0,368.0,03003}{0,02.10^{-3}} \\ &= 552,55 \text{ W/m}^2 \cdot ^\circ\text{C} \end{aligned}$$

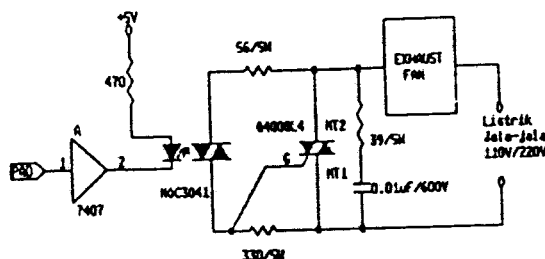
perpindahan kalor atau daya listrik yang diperlukan adalah (dari persamaan 4.3)

$$\begin{aligned} P &= h(\pi d)(T_w - T_\infty) \\ &= 552,55.(\pi.0,02.10^{-3})(107 - 29,6) \\ &= 2,687 \text{ W} \end{aligned}$$

Dalam perencanaan ini dipakai elemen pemanas strika 450 W.

#### IV.1.3. MODUL PENGGERAK KIPAS PENDINGIN

Rangkaian penggerak kipas pendingin merupakan rangkaian yang diaktifkan oleh triac dan tegangan jala-jala 220V akan terhubung dengan kipas pendingin yang akan digerakan. Kaki input dari IC 7407 dihubungkan dengan PPI 8255 dan kaki output



Gambar 4.9. Rangkaian Penggerak Kipas Pendingin

dihubungkan dengan optotriac MOC 3041 yang memiliki rangkaian detektor penyilang-nol didalamnya. Rangkaian tersebut dibuat aktif pada saat PPI 8255 memberikan masukan low.

Keluaran dari MOC 3041 dihubungkan dengan triac Q4008L4. Triac ini akan menghubungkan kipas pendingin dengan tegangan 220 V dari listrik jala-jala, pada saat optotriac aktif dan akan mentrigger triac tersebut pada kaki gate. Rangkaian penggerak kipas pendingin dapat dilihat pada gambar 4.9.

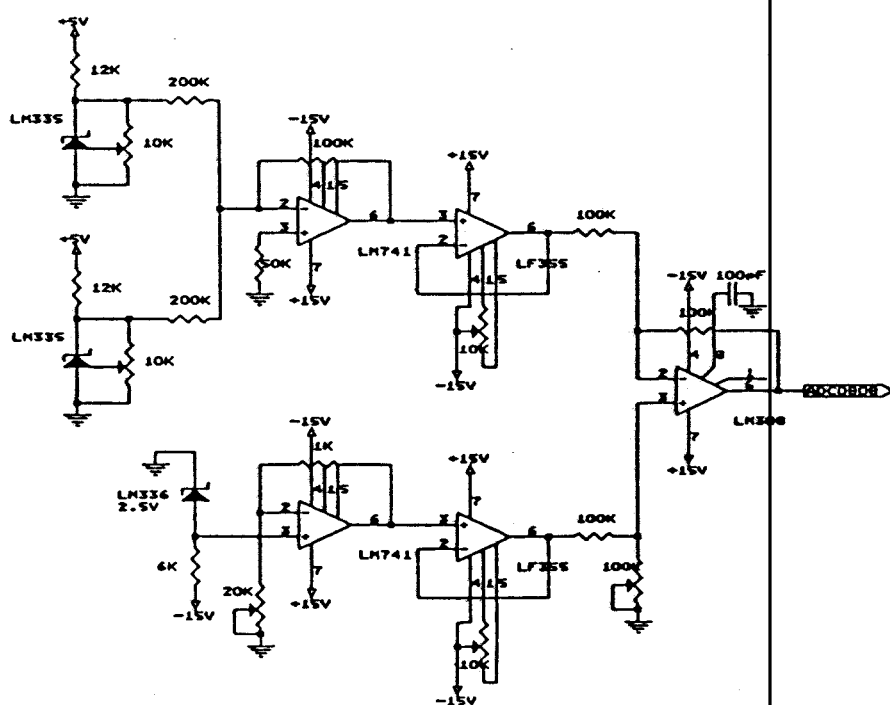
#### IV.1.4. MODUL SENSOR

##### RANGKAIAN SENSOR TEMPERATUR

Rangkaian sensor temperatur digunakan untuk mendeteksi temperatur ruang pemanas. Untuk sensor temperatur digunakan LM 335 yang merupakan IC sensor temperatur presisi buatan

National Semiconductor, sedangkan untuk rangkaian pengkondisi sinyal terdiri dari beberapa bagian : rangkaian buffer, rangkaian penguat perata-rata pembalik, rangkaian tegangan referensi dan rangkaian penguat instrumentasi. IC LM 335 mempunyai tegangan breakdown proporsional terhadap temperatur absolut sebesar  $10 \text{ mV}/^{\circ}\text{K}$ . Rangkaian pengkondisi sinyal temperatur ditunjukkan pada gambar 4.10.

Rangkaian penguat perata-rata pembalik dibutuhkan untuk mengambil nilai rata-rata dari dua titik pengukuran yang diambil oleh IC sensor temperatur LM 335. Dua titik pengukuran dibutuhkan agar nilai temperatur yang diambil dapat



Gambar 4.10. Rangkaian Pengkondisi Sinyal Temperatur

mewakili temperatur secara keseluruhan. Rangkaian penguat perata-rata pembalik ini disusun oleh op-amp LM 741. Tegangan keluaran dari penguat perata-rata pembalik dengan tiga tegangan masukan adalah sebagai berikut<sup>21)</sup> :

$$V_o = - \left[ \frac{E_1 + E_2 + E_3}{n} \right]$$

untuk dua tegangan masukan ( $n = 2$ ) diperoleh :

$$V_o = - \left[ \frac{E_1 + E_2}{2} \right]$$

dimana :

$$n = R/R_f$$

$$R = 2R_f$$

diambil  $R_f = 100 \text{ K}\Omega$ , maka  $R = 200 \text{ K}\Omega$ .

Karena sensor ini masih dalam satuan derajat kalvin, maka diubah kedalam derajat celcius, untuk itu diperlukan tegangan kalibrasi -2,73 Volt DC ( $0^\circ \text{C} = 273^\circ \text{K}$ ,  $273 \times 10 \text{ mV} = 2,73 \text{ Volt}$ ). Ini dihasilkan dari IC voltage reference LM 336 sebesar 2,5 Volt kemudian diperbesar oleh sebuah penguat tak-membalik sehingga diperoleh tegangan sebesar -2,73 Volt. Pengaturan pembesaran ini dilakukan dengan mengatur besar  $R_1$  dengan perhitungan sebagai berikut :

<sup>21)</sup> Robert F. Coughlin, Frederick F. Driscoll, op.cit, hlm. 41.

$$V_{ref} = A_{CL} \times (-2,5 \text{ Volt})$$

Jika  $V_{ref}$  yang diminta sebesar  $-2,73 \text{ Volt}$ , maka besarnya penguatan ( $A_{CL}$ ) adalah :

$$\begin{aligned} A_{CL} &= \frac{-2,73 \text{ V}}{-2,5 \text{ V}} \\ &= 1,092 \text{ Kali} \end{aligned}$$

Untuk penguatan rangkaian tak-membalik adalah<sup>22)</sup> :

$$A_{CL} = 1 + \frac{R_f}{R_1}$$

diambil  $R_f = 1 \text{ K}\Omega$ , maka :

$$\begin{aligned} R_1 &= \frac{R_f}{A_{CL} - 1} \\ &= \frac{1 \text{ K}\Omega}{1,092 - 1} \\ &= 10869,565 \Omega \end{aligned}$$

Agar diperoleh tegangan referensi  $-2,73 \text{ Volt}$  maka dipilih resistor variabel ( $R_1$ ) jenis multiturn  $15 \text{ K}\Omega$ .

Keluaran dari perata-rata tegangan dan tegangan referensi diumpankan ke penguat instrumentasi. Penguat instrumentasi ini dibutuhkan karena akuisisi data melewati transmisi line yang panjang. Penguat instrumentasi ini disusun

---

<sup>22)</sup> Ibid, hlm. 47.

oleh tiga op-amp. Keuntungan utama dari rangkaian ini adalah resistansi input yang tinggi. Penguat tegangan dari rangkaian tersebut sama dengan penguatan tegangan dari penguat diferensial satu op-amp.<sup>23)</sup>

$$V_o = m (E_1 - E_2) \quad (4.4)$$

dimana :

$V_o$  = tegangan output op-amp

$E_1$  = tegangan pada non-inverting input

$E_2$  = tegangan pada inverting input

$m$  = koefisien gain.

Persamaan 4.4 ini berlaku untuk kondisi sebagai berikut :

$$R_1 = R_2 = R$$

$$R_a = R_f = mR$$

$$m = mR/R$$

dimana :

$R_1$  = tahanan pada non-inverting input

$R_2$  = tahanan pada inverting input

$R_a$  = tahanan pada non-inverting input dengan ground

$R_f$  = tahanan feedback.

Penguatan dari penguat instrumentasi ini dibuat 1 kali ( $mR = R = 1 \text{ K}\Omega$ ) untuk mencegah tegangan input yang berlebihan ke ADC 0808. Agar sinyal yang didapat benar-benar merupakan sinyal yang akan diproses, maka pada buffer penguat instrumen-

---

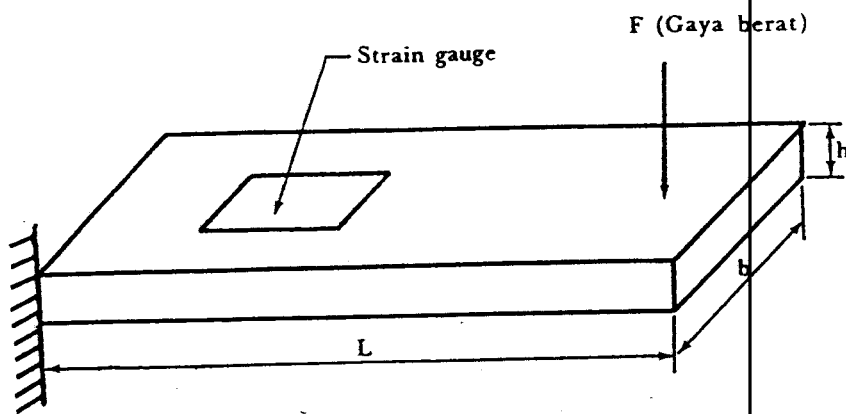
<sup>23)</sup> Ibid, hlm. 162.

tasi perlu diberi resistor pengatur tegangan offset. Untuk LF 355 diberi resistor offset jenis multiturn sebesar  $20k\Omega$ .

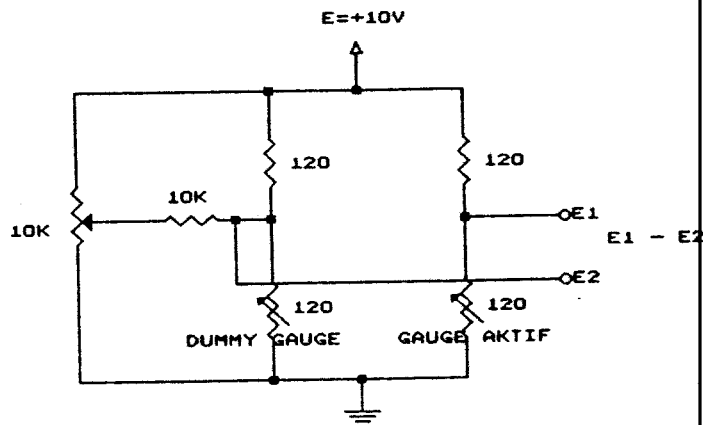
#### RANGKAIAN SENSOR BERAT

Untuk mengukur regangan yang terjadi pada specimen akibat gaya berat digunakan strain gauge. Gambar 4.11 memperlihatkan cara mengukur gaya berat dengan efek bending. Pada pengukuran gaya berat ini strain gauge diletakan pada ujung specimen yang diklam.

Pengukuran gaya berat ini digunakan jembatan wheatstone, dimana terdapat satu lengan aktif dan penerapan teknik kompensasi suhu dengan memasang dummy gauge. Maksud dari kompensasi suhu pada jembatan wheatstone adalah untuk mengeliminir perubahan temperatur yang terjadi pada gauge aktif.



Gambar 4.11. Pemasangan Strain Gauge pada Specimen



Gambar 4.12. Pemasangan Strain Gauge pada Jembatan Wheatstone

Pemasangan strain gauge pada jembatan wheatstone dengan rangkaian penyeimbang seperti gambar 4.12.

Pengukur gauge aktif dan dummy gauge pada gambar 4.12 memberikan perbedaan tegangan sebesar (persamaan 2.17) :

$$E_1 - E_2 = E \frac{\Delta R}{4R}$$

Tegangan keluaran diferensial jembatan  $E_1 - E_2$  masih terlalu kecil untuk dapat dikonversikan oleh pengubah analog ke digital. Untuk itu tegangan keluaran diferensial jembatan perlu dikuatkan, dan pada perencanaan ini digunakan penguat instrumentasi. Penguat instrumentasi ini dibutuhkan karena akuisisi data melewati transmisi line yang panjang dan resistansi input yang tinggi.



Untuk menentukan penguatan yang dibutuhkan, maka perlu diketahui data-data mekanis dari modul sensor ini :

■ SPECIMEN : Aluminium Campuran

- Lebar (b) = 15 mm
- Tebal (h) = 0,5 mm
- Panjang (L) = 100 mm
- E =  $74,5 \times 10^3 \text{ N/mm}^2$
- Tegangan leleh ( $\sigma_y$ ) =  $267,5 \text{ N/mm}^2$

■ STRAIN GAUGE

- Resistansi strain gauge = 120  $\Omega$
- Faktor gauge ( $S_g$ ) = 2,03

Dari data-data diatas, dicari regangan maksimum ( $\epsilon_{max}$ ) pada tegangan leleh ( $\sigma_y$ ) sesuai dengan persamaan 2.1 adalah :

$$\begin{aligned}\epsilon_{max} &= \frac{\sigma_y}{E} \\ &= \frac{267,5}{74,5 \cdot 10^3} \\ &= 3,6 \times 10^{-3}\end{aligned}$$

Pada perencanaan ini gaya berat maksimum yang diberikan pada specimen adalah 0,196 N (20 gr). Dari persamaan 2.5, regangan yang terjadi pada gaya berat 0,196 adalah :

$$\epsilon = \frac{6.F.L}{b.h^2.E}$$

$$\begin{aligned}
 &= \frac{6.0,196.100}{15.(0,5)^2.74,5.10^3} \\
 &= 4,2 \times 10^{-4}
 \end{aligned}$$

diperoleh  $\varepsilon < \varepsilon_{\max}$ , maka specimen masih berada pada daerah elastis. Dari persamaan 2.15, diperoleh :

$$S_g = \frac{\Delta R/R}{\varepsilon}$$

$$\begin{aligned}
 \Delta R/R &= (4,2 \times 10^{-4}).2,03 \\
 &= 8,5 \times 10^{-4}
 \end{aligned}$$

Dari persamaan 2.17, dimana  $E = 10 \text{ V}$  :

$$\begin{aligned}
 E_1 - E_2 &= E \frac{\Delta R}{4R} \\
 &= \frac{10 \cdot (8,5 \times 10^{-4})}{4} \\
 &= 2,125 \times 10^{-3}
 \end{aligned}$$

Sehingga dibutuhkan penguatan untuk gaya berat maksimum 0,196 N (20 gr) :

$$\frac{1,28}{2,125.10^{-3}} = 602,35 \text{ kali}$$

Dari perencanaan ini dibuat penguatan sebesar 602,35 kali dengan rangkaian penguat instrumentasi, hal ini karena keluaran dari jembatan wheatstone sangat kecil. Penguatan

tersebut dibagi menjadi dua step yaitu 60,235 kali dan 10 kali.

Penguat instrumentasi dibangun memakai tiga buah op-amp. Dua sebagai penyangga dan satu sebagai penguat diferensial, seperti yang terlihat pada gambar 4.13. Tahanan mR dapat diubah-ubah untuk menyeimbangkan setiap tegangan mode-bersama. Hanya ada dua tahanan, aR dan mR yang digunakan untuk menyetel gain sesuai dengan persamaan<sup>24)</sup> :

$$\frac{V_{01}}{E_1 - E_2} = 1 + \frac{2}{a} \quad (4.5)$$

$$\frac{V_{02}}{V_{01}} = \frac{mR}{R} \quad (4.6)$$

dimana  $a = aR/R$ .

Persamaan 4.5 untuk penguatan 60,235 kali, bila  $R = 10 \text{ K}\Omega$  maka :

$$60,235 = 1 + \frac{2}{a}$$

$$a = 33,764 \times 10^{-3}$$

kemudian

$$\begin{aligned} aR &= (33,764 \times 10^{-3}) \cdot (10 \times 10^3) \\ &= 337,64 \Omega \end{aligned}$$

---

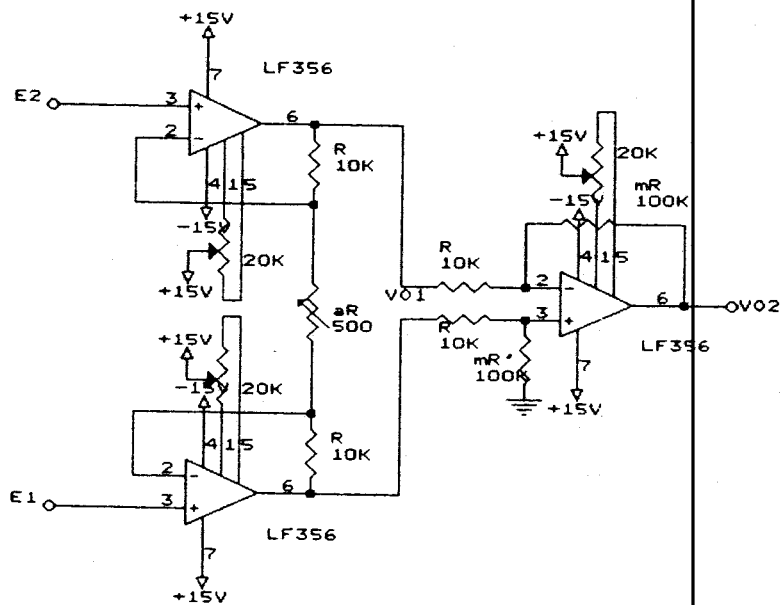
<sup>24)</sup> Ibid, hlm. 168.

Persamaan 4.6 untuk penguatan 10 kali :

$$10 = \frac{mR}{10 \times 10^3}$$

$$mR = 100 \text{ K}\Omega$$

Untuk tahanan  $337,64 \text{ }\Omega$  didapat dengan mengatur tahanan variabel  $500 \text{ }\Omega$  sehingga didapat penguatan  $60,235$  kali. Pada penguatan kedua didapat penguatan 10 kali dengan harga tahanan  $mR = 100 \text{ K}\Omega$  dan tahanan  $R = 10 \text{ K}\Omega$  sehingga total penguatan  $602,35$  kali. Semua tahanan yang dipakai mempunyai toleransi

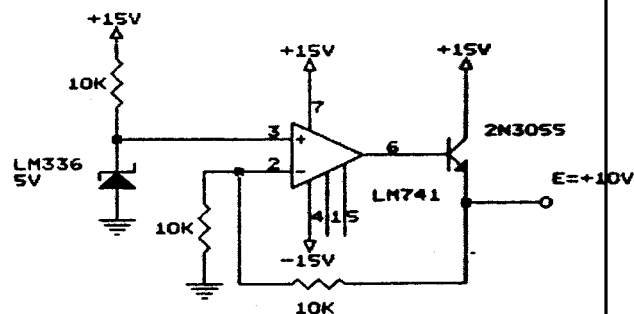


Gambar 4.13. Rangkaian Penguat Instrumentasi

1 % dan IC op-amp yang dipakai adalah jenis LF 356 jenis JFET yang mempunyai input impedansi sangat tinggi ( $10^{12} \Omega$ ). Gambar 4.13 menunjukkan rangkaian penguat instrumentasi dengan penguatan total 602,35 kali.

Gambar 4.14 merupakan tegangan eksitasi sebesar 10 volt dipasang pada ujung atas jembatan wheatstone, kestabilan tegangan ini dikontrol oleh sebuah diode referensi presisi 5,0 volt jenis LM 336-5,0 V pada masukan tak-membaliknya. Pengaturan tegangan sebesar 10 volt diatur dengan penguatan 2 kali pada penguat tak-membalik. Penguatan 2 kali didapat dengan harga tahanan  $R_i = 10 \text{ K}\Omega$  dan  $R_f = 10 \text{ K}\Omega$ .

Pengaturan penguat tak-membalik sanggup memberikan arus beban yang besar dengan menambahkan sebuah transistor pendorong arus seperti gambar 4.14.



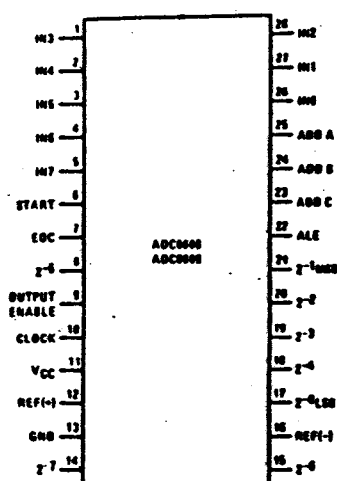
Gambar 4.14. Rangkaian Tegangan 10 Volt dengan Transistor Pendorong Arus

#### IV.1.5. MODUL ANALOG TO DIGITAL CONVERTER

##### RANGKAIAN ADC 0808

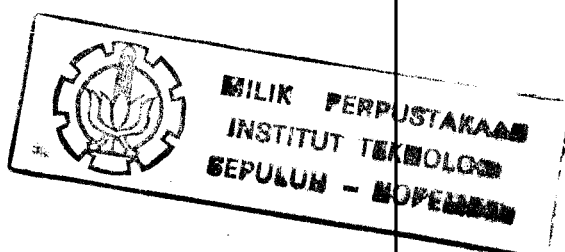
Dalam perencanaan alat ini ADC yang dipakai adalah ADC 0808 dimana jenis ADC ini dikelompokkan dalam tipe successive approximation ADC dan merupakan komponen CMOS dengan 8 bit analog to digital converter yang mempunyai 8 buah channel multiplexer. Gambar 4.15 menunjukkan ADC 0808.

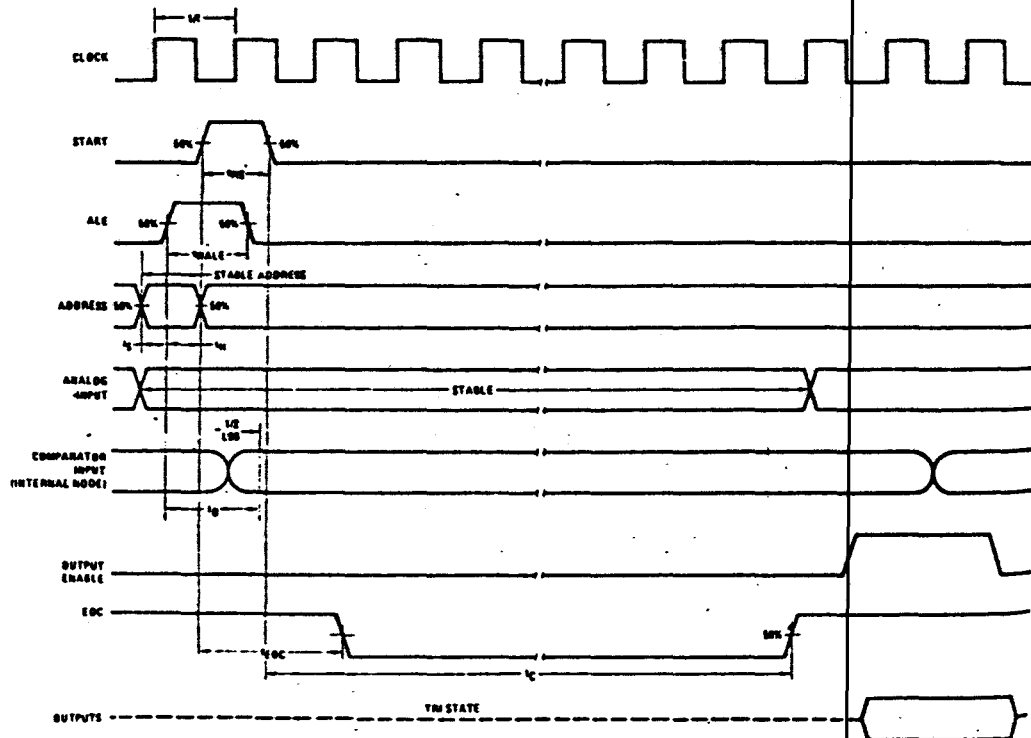
Untuk memilih channel yang dikehendaki dapat dipilih dengan memasukkan address channel ke address line input A, B dan C disertai sinyal dari low ke high pada ALE. Diagram waktu ADC dapat dilihat pada gambar 4.16.



Gambar 4.15. ADC 0808<sup>25)</sup>

<sup>25)</sup> CMOS Data Book, National Semiconductor Corporation, 1984, hlm. 5-50.





Gambar 4.16. Diagram Waktu ADC 0808<sup>26)</sup>

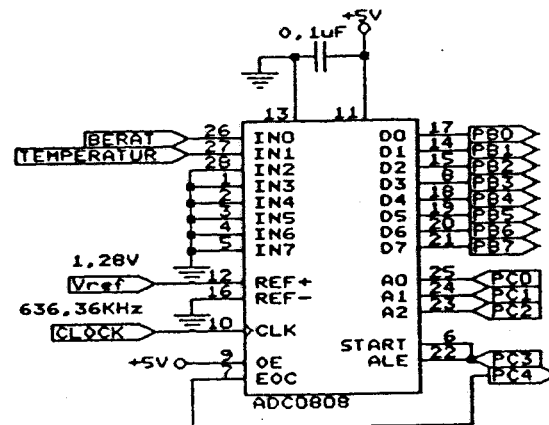
Penjelasan diagram waktu pada gambar 4.16 adalah sebagai berikut :

- sinyal clock, menentukan kecepatan proses
- sinyal START, memulai proses konversi
- sinyal ALE, memasukkan data address
- data address, menentukan channel analog yang aktif

<sup>26)</sup> Loc.cit.

- analog input, input sinyal analog yang akan dikonversikan
- komparator input, sinyal analog siap dikonversikan
- output enable (OE), gate yang memungkinkan hasil konversi dapat dikeluarkan dari ADC
- sinyal EOC, sinyal penandaan proses konversi selesai
- output, data digital hasil konversi.

Modul rangkaian sensor temperatur dan sensor berat dihubungkan terhadap input sinyal analog ADC 0808. Untuk memilih input analog yang akan dilihat nilai digitalnya, maka dilakukan dengan memilih jalur address bit yaitu A, B dan C. Jalur address ini diinputkan melalui port C PPI 8255 pada bit PC0, PC1 dan PC2. Kedelapan bit data digital hasil konversi



Gambar 4.17. Rangkaian ADC 0808



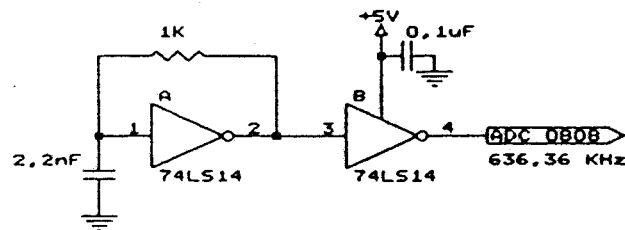
input analog dari sensor dibaca melalui port B PPI 8255. Sinyal kontrol START dan ALE diatur oleh software melalui port C PPI 8255 bit PC3, sedangkan sinyal output kontrol EOC dimonitor oleh port C PPI 8255 bit PC4. Gambar 4.17 menunjukkan rangkaian ADC 0808.

Pada clock cycle yang pertama diinputkan address channel berupa address 3 bit dengan kombinasi bit tergantung pada input analog mana yang hendak dikonversikan. Lalu serentak pin ALE dan START diberi logika '1', saat transisi dari logika '1' ke logika '0' proses konversi berlangsung. Pada saat konversi selesai, keluaran EOC akan berubah dari logika '0' menuju logika '1', hal ini memberitahukan mikrokomputer bahwa ADC 0808 telah selesai melakukan konversi.

#### RANGKAIAN CLOCK

Rangkaian pembangkit clock ADC 0808 dibentuk dari sebuah schmitt trigger inverter 74LS14 dari rangkaian resistor kapasitor. Gambar 4.18 menunjukkan gambar rangkaian clock. Frekuensi clock yang dihasilkan 640 KHz.

Cara kerja rangkaian ini adalah sebagai berikut : misal mula-mula input inverter pertama adalah '1', dan output '0', maka kapasitor akan discharge lewat tahanan ke output inverter. Tegangan input inverter pertama turun sampai menuju harga logika '0', akibatnya output inverter akan '1'.

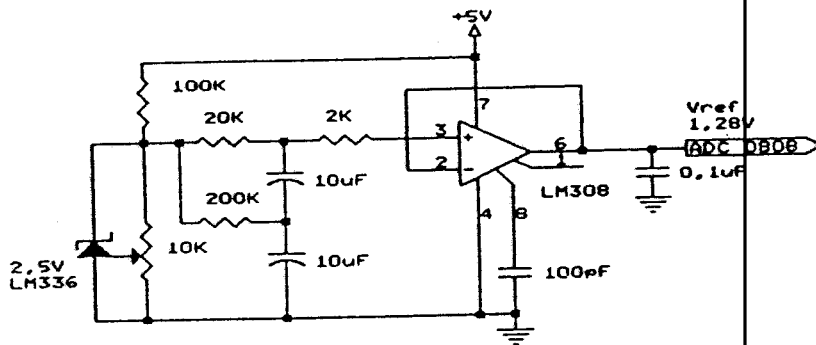


Gambar 4.18. Rangkaian Clock ADC 0808

Kemudian proses akan berulang kembali seterusnya. Untuk memperoleh bentuk gelombang clock yang cukup baik (square) maka dilewatkan ke inverter ke dua, baru kemudian sinyal clock ini diinputkan ke input clock ADC 0808.

#### RANGKAIAN TEGANGAN REFERENSI

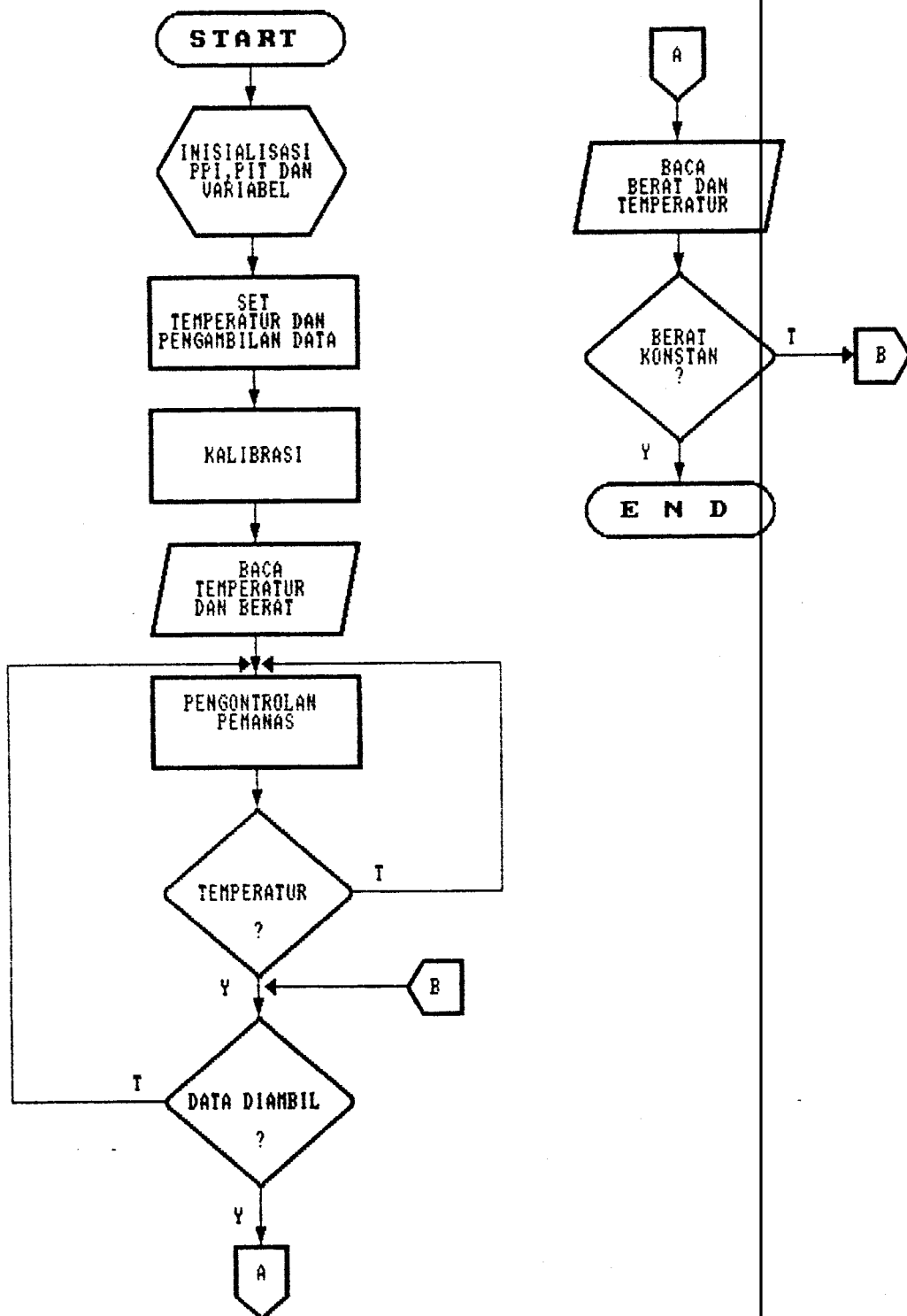
Rangkaian penghasil tegangan referensi 1,28 Volt untuk input tegangan referensi ADC ini dihasilkan oleh sebuah dioda referensi presisi 2,5 Volt teregulasi jenis IC LM 336-2,5 V, yang kemudian dibagi oleh rangkaian resistor pembagi tegangan sehingga diperoleh tegangan sekitar 1,28 Volt dibuffer oleh LM 308 sebelum diumpankan ke input REF+ dan REF- ADC 0808. Gambar 4.19 menunjukkan rangkaian tersebut.



Gambar 4.19. Rangkaian Tegangan Referensi 1,28 Volt

#### IV.2. PERENCANAAN SOFTWARE

Untuk menjalankan peralatan ini dengan bantuan mikrokomputer IBM PC-XT diperlukan bahasa pemrograman. Bahasa pemrograman yang digunakan adalah bahasa aras tengah (mid level language) yaitu Turbo C. Turbo C memiliki efisiensi sintaks naskahnya dan efektivitasnya tinggi karena secara ekstensif menggunakan pointer. Penggunaan pointer secara ekstensif menjadikan bahasa turbo C sangat dekat dengan bahasa Assembly, demikian pula fasilitas yang dikandung sejajar dengan bahasa aras atas lain yang manapun.



Gambar 4.20. Flowchart dari Program yang dibuat secara Garis Besar



Logika pemrograman yang direncanakan seperti terlihat pada flowchart gambar 4.20. Mula-mula menginisialisasi PPI 8255 dengan menset control word register. Dalam hal ini PPI 8255 dioperasikan pada mode 0 dengan port A dan port C lower digunakan sebagai output, port B dan port C upper digunakan sebagai input. Control word register yang dikirim adalah :

```
outportb(0x307,0x8A);
```

dimana 0x8A adalah control word register dan 0x307 alamat dari control word tersebut. Kemudian menginisialisasi PIT 8253 yang dioperasikan pada mode 1 counter 0. Control word register yang dikirim adalah :

```
outportb(0x30B,0x13);
```

dimana 0x30B adalah alamat dari control word tersebut. Setelah dilakukan inisialisasi, maka temperatur dan pengambilan data di set sesuai dengan kebutuhan.

Sebelum dilakukan pembacaan berat, timbangan harus selalu dalam keadaan standard. Dimana setiap saat timbangan perlu dikalibrasi dengan batu timbangan yang telah diketahui beratnya. Setelah itu dilakukan pembacaan berat sampel batubara pada temperatur kamar.

Langkah selanjutnya penyalaan pemanas untuk mencapai temperatur kerja berkisar 104 °C sampai 110 °C dengan kontrol proporsional. Selama dipanaskan pada temperatur kerja, selang waktu pembacaan berat dapat di-set. Pembacaan berat ini terus dilakukan sampai diperoleh pengukuran berat yang konstan

dengan terus mengontrol penyalan pemanas. Setelah diperoleh berat konstan maka proses ini selesai dan data hasil pembacaannya untuk selanjutnya dapat disimpan pada diskette atau dicetak pada printer.

#### MENGATUR KERJA PERANGKAT KERAS

Perangkat keras yang dibuat akan bekerja sesuai dengan yang diharapkan, apabila semua perintah yang diberikan sesuai dengan prosedur kerja dari komponen tersebut. IC-IC yang perlu pengaturan adalah PPI 8255, PIT 8253 dan ADC 0808.

PPI 8255 akan mengatur kerja dari ADC 0808 dan penggerak relay pengatur kipas pendingin dan PIT 8253 menghasilkan sinyal PWM untuk mengatur kerja pemanas secara kontinyu. Pengaturan kerja PPI 8255 dan PIT 8253 sudah dijelaskan diatas.

Untuk pengambilan data pengukuran dari sensor gaya berat dan sensor temperatur digunakan ADC 0808. Berikut ini program yang digunakan untuk pengambilan data :

```
unsigned char sensor(unsigned char channel)
{
    unsigned char EOC,data;

    outportb(PC,channel);
    outportb(PC,channel!0x80);
    outportb(PC,channel);
    while(EOC) EOC=inportb(PC)&0x10;
    while(!EOC) EOC=inportb(PC)&0x10;

    data=inportb(PB);
    return(data);
}
```

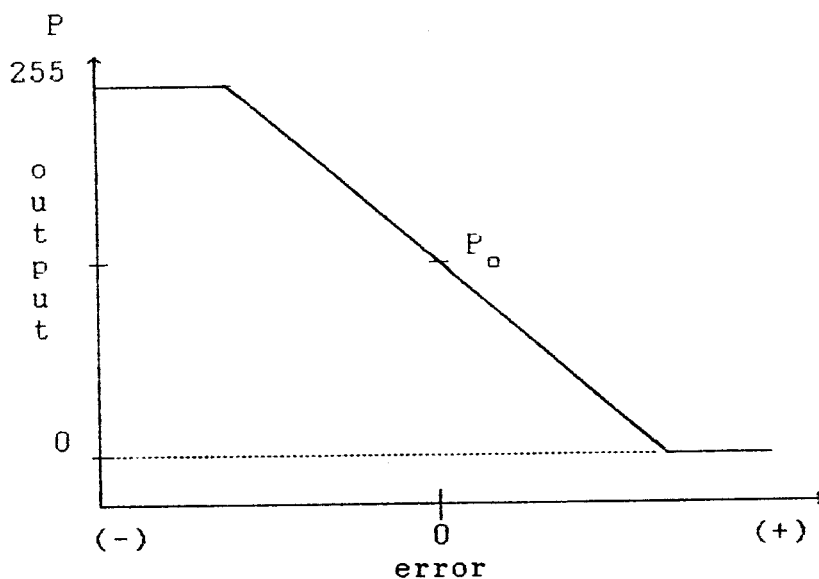
Sinyal analog yang telah diubah menjadi sinyal digital diolah oleh mikrokomputer.

Jenis kontrol yang dipilih untuk pengaturan pemanas adalah proporsional. Algoritma untuk kontrol proporsional sebagai berikut :<sup>27)</sup>

$$P = K_p \cdot E_p + P_o \quad (4.7)$$

dimana :

$K_p$  : Konstanta proporsional



Gambar 4.21. Diagram Kontrol Proporsional yang dibuat

<sup>27)</sup> Curtis D. Johnson, Process Control Instrumentation Technology, John Wiley Sons Inc., Canada, 1988, hlm. 359.

$E_p$  : Error dalam persen

$P_o$  : Output kontroler tanpa error.

Pada saat selisih antara temperatur sebenarnya dengan setting titik kerja sama dengan nol ( $\text{error}=0$ ) maka persamaannya sebagai berikut :

$$P_o = 255 - K_p \cdot \text{set\_point} \quad (4.8)$$

Persamaan 4.8 ini dibuat dengan asumsi panas yang dilepas oleh pemanas harus seimbang dengan panas yang diserap oleh benda disekitarnya. Untuk mendapatkan aksi kontrol dengan overshoot yang kecil, maka persamaan kontrol yang dibuat dalam persamaan eksponensial.

■ Pada saat ( $\text{error} < 0$ ) maka persamaan yang dibuat adalah

$$\text{PWM} = 255 \cdot (1 - \exp(K_p \cdot \text{error})) \quad (4.9)$$

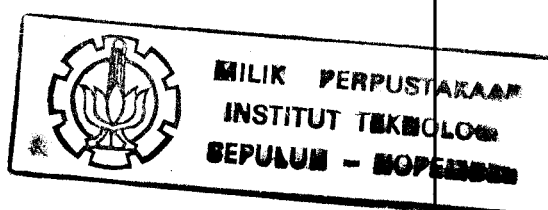
■ Pada saat ( $\text{error} > 0$ ) maka persamaan yang dibuat adalah

$$\text{PWM} = P_o + 255 \cdot \exp(-\text{abs}(K_p \cdot \text{error})) \quad (4.10)$$

Selain dari pada itu persamaan diatas juga memper-timbangkan pada saat menurunkan temperatur dibutuhkan kondisi yang ekstrim dengan langsung mematikan pemanas.

Besarnya nilai  $K_p$  dapat ditentukan sampai dicapai sistem yang paling mendekati keadaan ideal dari sistem yang diharapkan.

Program kontrol pemanas secara keseluruhan adalah sebagai berikut :





```
int control()
{
    error=set_point-var_temp;
    Po=255-(Kp*set_point);
    if(error<0) PWM=255*(1-exp(Kp*error));
    if(error=0) PWM=Po;
    if(error>0) PWM=Po+255*exp(-abs(Kp*error));
    return(PWM);
}

void proses()
{
    .....;
    .....;
    PWM=control();
    outportb(COUNT1,PWM);
    .....;
    .....;
}
```

## BAB V

### PENGUJIAN DAN PENGUKURAN PERALATAN

---

Beberapa unit dari rangkaian elektronis dalam sistem tersebut memerlukan kalibrasi dan pengujian terhadap operasi rangkaian. Terutama sensor temperatur dan sensor berat perlu dikalibrasi agar penunjukkan pembacaan sensor memang benar-benar merupakan besaran yang diukur. Sedangkan rangkaian yang memerlukan pengujian adalah rangkaian pengontrol pemanas.

#### V.1. PENGUJIAN SENSOR TEMPERATUR

Rangkaian sensor temperatur dikalibrasi pada temperatur kamar  $29,6^{\circ}\text{C}$ , karena IC LM 335 direkomendasikan memiliki linieritas yang baik maka dengan satu titik kalibrasi akan berlaku untuk semua temperatur<sup>28)</sup>. Hasil pengujian terhadap berbagai temperatur dengan membandingkan antara hasil pembacaan sensor dengan thermometer air raksa ditabelkan pada tabel 5.1.

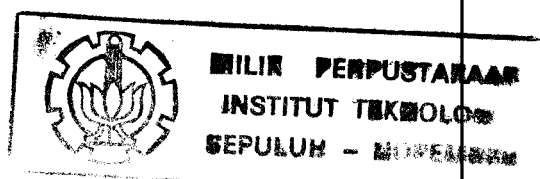
---

<sup>28)</sup> Charles A. Schuler, William L. McNamee, op.cit. hlm. 218.

Tabel 5.1. Hasil Pengujian Sensor Temperatur

| No. | Thermometer<br>Air Raksa (T)<br>( ° C ) | Tampilan pada<br>Monitor ( ° C ) |
|-----|---|----------------------------------|
| 1   | 29,6                                    | 29,6                             |
| 2   | 40                                      | 40,2                             |
| 3   | 50                                      | 49,9                             |
| 4   | 60                                      | 60,1                             |
| 5   | 70                                      | 70,2                             |
| 6   | 80                                      | 80,2                             |
| 7   | 90                                      | 90,1                             |
| 8   | 100                                     | 100,0                            |

Karena temperatur maksimum yang dapat diukur oleh IC LM 335 adalah 100 °C sedangkan pada perencanaan alat ini dibutuhkan pengukuran temperatur berkisar 107 °C. Untuk mengatasi hal tersebut sensor LM 335 diberi isolasi untuk mengurangi (menghambat) panas yang diterima. Hasil pengukuran terhadap berbagai temperatur dengan membandingkan antara hasil pembacaan sensor dengan thermometer air raksa ditabelkan pada tabel 5.2.



Tabel 5.2. Hasil Pengukuran Sensor Temperatur Setelah disolasi

| No. | Thermometer<br>Air Raksa (T)<br>( $^{\circ}\text{C}$ ) | Tampilan pada<br>Monitor (TM)<br>( $^{\circ}\text{C}$ ) |
|-----|--|---|
| 1   | 29,6   | 30  |
| 2   | 40   | 36  |
| 3   | 50   | 42  |
| 4   | 60   | 49  |
| 5   | 70   | 56  |
| 6   | 80   | 63  |
| 7   | 90   | 70  |
| 8   | 100  | 77  |
| 9   | 107  | 82  |
| 10  | 110  | 85  |

Dengan pendekatan regresi linier akan diperoleh hubungan antara hasil pembacaan sensor temperatur dengan temperatur sebenarnya.

$$T = a_0 TM + a_1$$

dimana :

$$a_0 = \frac{N(\sum TM.T) - (\sum TM)(\sum T)}{N(\sum TM^2) - (\sum TM)^2}$$

$$a_1 = \frac{(\sum T)(\sum TM^2) - (\sum TM)(\sum TM.T)}{N(\sum TM^2) - (\sum TM)^2}$$

dari tabel 5.2 diperoleh

$$N = 10$$

$$\Sigma TM = 590$$

$$\Sigma T = 736,6$$

$$\Sigma TM^2 = 38244$$

$$\Sigma TM.T = 48452$$

maka diperoleh :

$$a_0 = 1,45$$

$$a_1 = - 12,12$$

jadi persamaan regresi liniernya adalah :

$$T = 1,45 TM - 12,12$$

dimana :

$$TM = (\text{data digital}) \times (V_{ref}/255) \times 100$$

Dari hubungan diatas diperlukan untuk mendapatkan pembacaan temperatur dalam derajat celcius melalui sistem software. Hasil pengujian sensor temperatur dengan pendekatan regersi linier ditabelkan pada tabel 5.3.

Tabel 5.3. Hasil Pengujian Sensor Temperatur dengan Pendekatan Regresi Linier

| No. | Thermometer<br>Air Raksa (T)<br>( °C ) | Tampilan<br>pada Monitor<br>( °C ) |
|-----|--|------------------------------------|
| 1   | 29,6                                   | 30                                 |
| 2   | 40                                     | 41                                 |
| 3   | 50                                     | 50                                 |
| 4   | 60                                     | 60                                 |
| 5   | 70                                     | 71                                 |
| 6   | 80                                     | 81                                 |
| 7   | 90                                     | 91                                 |
| 8   | 100                                    | 101                                |
| 9   | 107                                    | 109                                |
| 10  | 110                                    | 112                                |

## V.2. PENGUJIAN SENSOR BERAT

Sensor berat menggunakan strain gauge yang diletakan pada ujung specimen yang diklam. Prinsip dasar dari pengukuran berat ini adalah mengukur regangan yang terjadi pada specimen akibat gaya yang bekerja pada ujung specimen yang tidak diklam. Sesuai kurva tegangan dan regangan (lihat gambar 2.13), perubahan tegangan (gaya per satuan luas) pada specimen berbanding lurus dengan perubahan regangan pada daerah elastis. Setelah dilakukan pengukuran tegangan output dari pengkondisi sinyal sensor berat dengan membandingkannya dengan batu timbangan ditabelkan pada tabel 5.4.

Tabel 5.4. Hasil Pengukuran Tegangan Output Pengkondisi Sinyal

| No. | Berat (B)<br>(gram) | Tegangan Output (Vo)<br>Pengkondisi Sinyal (Volt) |          |
|-----|---------------------|---|----------|
|     |                     | T=29,6 °C   | T=107 °C |
| 1   | 0,0                 | 0,000   | 0,000    |
| 2   | 2,0                 | 0,112   | 0,107    |
| 3   | 4,0                 | 0,196   | 0,201    |
| 4   | 6,0                 | 0,312   | 0,300    |
| 5   | 8,0                 | 0,416   | 0,406    |
| 6   | 10,0                | 0,517   | 0,511    |
| 7   | 12,0                | 0,605   | 0,601    |
| 8   | 14,0                | 0,698   | 0,702    |
| 9   | 16,0                | 0,801   | 0,812    |
| 10  | 18,0                | 0,910   | 0,907    |
| 11  | 20,0                | 1,280   | 1,278    |

Dengan pendekatan regresi linier diperoleh hubungan berat dengan tegangan output sebagai berikut :

$$B = a V_o + b$$

dimana :

$$a = \frac{N(\sum V_o \cdot B) - (\sum V_o)(\sum B)}{N(\sum V_o^2) - (\sum V_o)^2}$$

$$b = \frac{(\sum B)(\sum V_o^2) - (\sum V_o)(\sum V_o \cdot B)}{N(\sum V_o^2) - (\sum V_o)^2}$$

Dari tabel 5.4 diperoleh :

pada T = 29,6 °C

$$\begin{aligned} N &= 11 \\ \sum V_o &= 5,847 \\ \sum B &= 110,0 \\ \sum V_o^2 &= 4,549 \\ \sum V_o \cdot B &= 83,206 \end{aligned}$$

maka diperoleh :

$$\begin{aligned} a &= 17,165 \\ b &= 0,876 \end{aligned}$$

pada T = 107 °C

$$\begin{aligned} N &= 11 \\ \sum V_o &= 5,825 \\ \sum B &= 110,0 \\ \sum V_o^2 &= 4,537 \\ \sum V_o \cdot B &= 83,094 \end{aligned}$$

maka diperoleh :

$$\begin{aligned} a &= 17,106 \\ b &= 0,842 \end{aligned}$$

persamaan regresi liniernya adalah :

pada T = 29,6 °C :  $B = 17,165 V_o + 0,876$



pada  $T = 107^{\circ}\text{C}$  :  $B = 17,106 V_o + 0,942$

Dari hubungan diatas diperlukan untuk mendapatkan pembacaan berat dalam gram melalui sistem software, yaitu :

pada  $T = 29,6^{\circ}\text{C}$  :

$$B = 17,165 V_o + 0,876$$

$$V_o = (\text{digital word}) \times (V_{\text{ref}}/255)$$

maka :

$$B = 17,165 \times ((\text{digital word}) \times (V_{\text{ref}}/255)) + 0,876$$

pada  $T = 107^{\circ}\text{C}$  :

$$B = 17,106 V_o + 0,942$$

$$V_o = (\text{digital word}) \times (V_{\text{ref}}/255)$$

maka :

$$B = 17,106 \times ((\text{digital word}) \times (V_{\text{ref}}/255)) + 0,942$$

Hasil pengujian sensor berat ditabelkan pada tabel 5.5.



Tabel 5.5. Hasil Pengujian Sensor Berat

| No. | Berat<br>(Gram) | Tampilan pada Monitor<br>(Gram) |             |
|-----|-----------------|---------------------------------|-------------|
|     |                 | T=29,6 °C                       | T=107 °C    |
| 1   | 0,0             | 0,0                             | 0,0         |
| 2   | 2,0             | 2,0 - 2,6                       | 2,0 - 2,4   |
| 3   | 4,0             | 4,0 - 4,5                       | 4,0 - 4,5   |
| 4   | 6,0             | 6,0 - 6,7                       | 6,0 - 6,6   |
| 5   | 8,0             | 8,0 - 8,4                       | 8,0 - 8,5   |
| 6   | 10,0            | 10,0 - 10,6                     | 10,0 - 10,4 |
| 7   | 12,0            | 12,0 - 12,4                     | 12,0 - 12,5 |
| 8   | 14,0            | 14,0 - 14,6                     | 14,0 - 14,4 |
| 9   | 16,0            | 16,0 - 16,5                     | 16,0 - 16,5 |
| 10  | 18,0            | 18,0 - 18,5                     | 18,0 - 18,5 |
| 11  | 20,0            | 20,0 - 20,0                     | 20,0 - 20,0 |

### V.3. PENGUJIAN TERHADAP RANGKAIAN UMPAN-BALIK PROPOSIONAL

Kontrol proposional pada peralatan ini dimungkinkan karena adanya rangkaian umpan-balik yang disusun oleh PIT 8253, rangkaian sinkronisasi, optotriac dan triac. Gambar 5.1. memperlihatkan sinyal sinkronisasi level TTL yang dihasilkan oleh rangkaian zero-crossing detector.

### V.4. PENENTUAN KANDUNGAN AIR DALAM BATUBARA

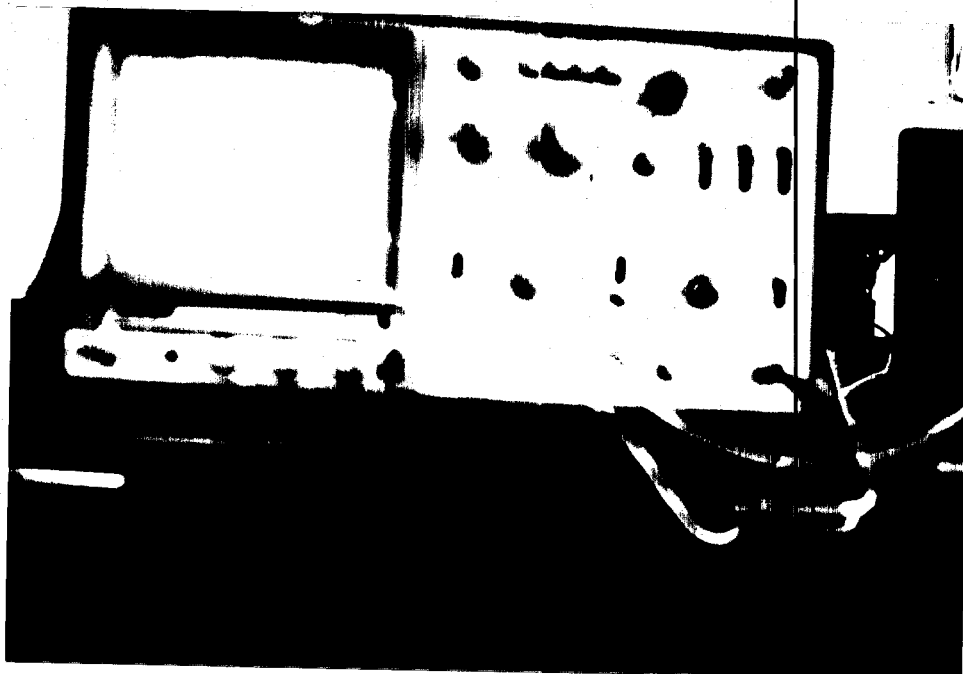
Hasil penentuan kandungan air dalam batubara dengan peralatan yang dibuat ini dibandingkan dengan cara konvensional ditabelkan pada tabel 5.6. Hasil penentuan kandungan air dalam batubara untuk satu pengukuran sampel dapat dilihat pada gambar 5.3.

Tabel 5.6. *Penentuan Kandungan Air Dalam Batubara*

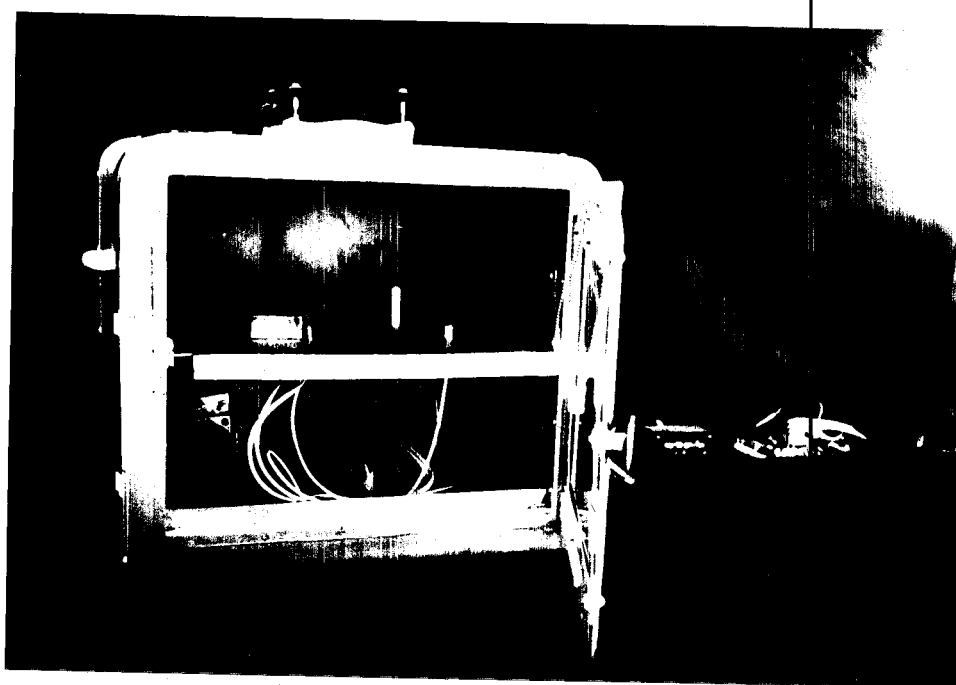
| CARA<br>KONVEN-<br>SIONAL (%) | DENGAN PERALATAN YANG DIBUAT (%) |          |           |                                     |
|-------------------------------|----------------------------------|----------|-----------|-------------------------------------|
|                               | PENG. I                          | PENG. II | PENG. III | RATA-RATA                           |
| 1,30                          | 1,27                             | 1,32     | 1,25      | $\frac{1,27+1,32+1,25}{3}$ $= 1,28$ |

CATATAN :

PENG. I = PENGUKURAN PERTAMA  
 PENG. II = PENGUKURAN KEDUA  
 PENG. III = PENGUKURAN KETIGA



Gambar 5.1. *Sinyal Sinkronisasi*



Gambar 5.2. *Prototipe Peralatan Yang Dibuat*

| MOISTURE: SOLVER |              |
|------------------|--------------|
| Nama File        | : BARA1      |
| Tgl/bln/thn      | : 4/ 7/1993  |
| Jam              | : 8:45:41    |
| Temperatur Awal  | : 30 °C      |
| Berat Awal       | : 18.89 gram |
| Temperatur Akhir | : 106 °C     |
| Berat Akhir      | : 18.65 gram |
| Moisture         | : 1.27 %     |

Gambar 5.3. Hasil Pengukuran Dicetak Pada Printer

## BAB VI

### KESIMPULAN

---

#### VI.1. KESIMPULAN

Kesimpulan yang dapat ditarik dari pembuatan tugas akhir ini adalah sebagai berikut :

1. Unjuk-kerja dari peralatan yang dibuat setelah dilakukan pengujian dan pengukuran adalah :
  - Pengukuran oleh sensor temperatur pada sistem tersebut, dalam pengujian dengan rentang pengukuran  $29,6^{\circ}\text{C}$  sampai dengan  $107^{\circ}\text{C}$  memiliki kesalahan rata-rata  $0,94^{\circ}\text{C}$ . Maka pengukuran temperatur tersebut memiliki ketelitian sebesar  $1,17\%$ .
  - Pengukuran oleh sensor berat pada sistem tersebut, dalam pengujian pada rentang  $0$  gram sampai dengan  $20$  gram pada temperatur  $29,6^{\circ}\text{C}$  dan  $107^{\circ}\text{C}$  memiliki kesalahan rata-rata  $0,73$  gram. Maka pengukuran berat pada sistem tersebut memiliki ketelitian sebesar  $3,6\%$ .
2. Ketelitian pengukuran temperatur relatif masih dalam rentang pengukuran yang diijinkan, sedangkan untuk pengukuran berat kurang teliti hal ini disebabkan karena sinyal yang diukur sangat kecil.
3. Specimen menentukan sensitifitas dari pengukuran berat.

4. Dengan kontrol proporsional pada penyalan pemanas memberikan keunggulan dalam kecermatan untuk mencapai harga temperatur yang diinginkan.
5. Fasilitas software dilengkapi dengan menyimpan/mengambil data pada diskette dan mencetak pada printer sehingga memudahkan operator untuk menganalisa data tersebut.

## VI.2. SARAN-SARAN

Berikut ini diberikan saran-saran yang diharapkan dapat berguna untuk pengembangan lebih lanjut :

1. Untuk meningkatkan ketelitian dari pengukuran berat maka dilakukan dengan cara sebagai berikut :
  - pemasangan sensor berat (strain gauge) yang tepat pada specimen.
  - Power supply unit harus stabil (faktor ripple-nya kecil) terutama untuk mensupply jembatan wheatstone.
2. Selain untuk menentukan kandungan air, alat ini dapat dikembangkan untuk menentukan volatile matter, ash dan fixed carbon dalam batubara (pada analisa proksimat) dengan memilih sensor temperatur yang dapat mendeteksi temperatur 1000 °C dan sensor berat yang lebih sensitif, teliti dan tidak terpengaruh pada perubahan temperatur.

## DAFTAR PUSTAKA

---

1. Cooper, William D., Sahat Pakpahan (Penerjemah), INSTRUMENTASI ELEKTRONIK DAN TEKNIK PENGUKURAN, Edisi ke-2, Penerbit Erlangga, Jakarta, 1985.
2. Coughlin, Robert F., and Frederick F. Driscoll, Herman W. Soemitro (Penerjemah), PENGUAT OPERASIONAL DAN RANGKAIAN TERPADU, Penerbit Erlangga, Jakarta, 1985.
3. Griswold, John, FUELS, COMBUSTION AND FURNACES, McGraw-Hill Book Company Inc., New York, 1946.
4. Hall, Douglas V., MICROPROCESSORS AND INTERFACI PROGRAMMING AND HARDWARE, McGraw-Hill Inc., Singapore, 1988.
5. Hoffmann, FUNDAMENTALS OF STRAIN GAUGE TECHNIQUES, Hottinger Baldwin Messtechnik, 1974.
6. Holman, J.P., E. Jasjfi (Penerjemah), PERPINDAHAN KALOR, Edisi keenam, Penerbit Erlangga, Jakarta, 1988.
7. Johnson, Curtis D., PROCESS CONTROL INSTRUMENTATION TECHNOLOGY, John Willey Sons Inc., Canada, 1988.
8. Liu, Yu Cheng, and Glenn A. Gibson, MICROCOMPUTER SYSTEMS : 8086/8088 FAMILY, Prentice-Hall International, 1986.
9. Maloney, Timothy J., INDUSTRIAL SOLID-STATE ELECTRONICS, 2-nd, Prentice-Hall Inc., New Jersey, 1986.
10. Probststein, Ronald F., and R. Edwin Hicks, SYNTHETIC FUELS,



1-st, McGraw-Hill International Book Company, 1982.

11. Steeman, J.P.M., DATA SHEET BOOK 2, PT. Elex Media Komputindo, Jakarta, 1988.

12. Schuler, Charles A., and William L. McNamee, INDUSTRIAL ELECTRONICS AND ROBOTICS, 1-st, McGraw-Hill, Singapore, 1986.

13. Tocci, Ronald J., DIGITAL SYSTEMS, Prentice Hall Inc., 1980.

14. ...., LINEAR DATABOOK, National Semiconductor Corp., Santa Clara, California, 1982.

15. ...., CMOS DATABOOK, National Semiconductor Corp., 1984.

```

/* File : MOISTURE.C */

/*****
/*
/* SOFTWARE UNTUK PENENTUAN KANDUNGAN AIR */
/*          DALAM BATUBARA          */
*****/
/*  NAMA : I KOMANG WIDNYANA KARANG  */
/*  NRP  : 2882200945                */
*****/

#include "b:declares.h"
#include "b:procedur.c"

/*----- INFO -----*/
about2()
{
    int a;
    frame(13,8,66,17,79,0,LEFT_UP2,RIGTH_UP2,LEFT_DWN2,RIGTH_DWN2,
          HORIZONTAL2,VERTICAL2);
    write_string(33,8," TUGAS AKHIR ",79);
    for(a=0;a<=7;a++)
        write_string(14,a+9,about[a],79);
    while(getch()!=ESC);
}

int about1()
{
    char puzh;

    union inkey{
        char ch[2];
        int i;
    } c;

    frame(1,3,11,5,23,0,195,194,LEFT_DWN1,RIGTH_DWN1,
          HORIZONTAL1,VERTICAL1);
    write_string(2,4," About ",94);
    paint(1,25,80,25,17,0);
    write_string(2,25,"Esc",27);
    write_string(5,25,"-to Main Menu",30);
    do{
        puzh=getch();
        while(puzh==ENTER){
            gettext(13,8,67,19,about2_scr);about2();
            puttext(13,8,67,19,about2_scr);break;
        }
    }

```

```

    }while(puzh!=ESC);
}

/*----- FILE -----*/
int go_file()
{
    char puzh;
    int choise_f;

    choise_f=var_choise_f;
    hl(10,choise_f+3,94,9);
    do
    {
        puzh=getch();
        hl(10,var_choise_f+3,27,2);
        hl(12,var_choise_f+3,30,7);
        switch(puzh)
        {
            case DOWN :choise_f++;if(choise_f>2)choise_f=1;
                        var_choise_f=choise_f;break;
            case UP   :choise_f--;if(choise_f<1)choise_f=2;
                        var_choise_f=choise_f;break;
            case 'S'  :choise_f=1;var_choise_f=1;puzh=ENTER;break;
            case 's'  :choise_f=1;var_choise_f=1;puzh=ENTER;break;
            case 'L'  :choise_f=2;var_choise_f=2;puzh=ENTER;break;
            case 'l'  :choise_f=2;var_choise_f=2;puzh=ENTER;break;
            case ESC  :choise_f=0;puzh=ENTER;break;
        }
        hl(10,var_choise_f+3,94,9);
    }
    while(puzh!=ENTER);
    return(choise_f);
}

void masukan_nama_file(char *nama_file)
{
    frame(10,6,36,8,31,0,LEFT_UP1,RIGTH_UP1,LEFT_DWN1,RIGTH_DWN1,
        HORIZONTAL1,VERTICAL1);
    paint(1,25,80,25,17,0);
    write_string(2,25,"Esc",27);
    write_string(5,25,"-to Main Menu",30);

    write_string(11,7," Nama File : ",23);
    goxy(25,7);
    textbackground(BLUE);
    textcolor(WHITE);
    while((*nama_file=getche())!='\r') {++nama_file;}
    *(nama_file)='\0';
}

```

```

}
```

```

open_file_read(char *nama_file)
{
    if((pointer_ke_file=fopen(nama_file,"rb+"))==NULL) return 0;
    else return 1;
}
```

```

open_file_write(char *nama_file)
{
    if((pointer_ke_file=fopen(nama_file,"wb+"))==NULL) return 0;
    else return 1;
}
```

```

void rekam_data()
{
    int i;

    getdate(&d);
    gettime(&t);

    hari=d.da_day;
    bulan=d.da_mon;
    tahun=d.da_year;
    jam=t.ti_hour;
    menit=t.ti_min;
    detik=t.ti_sec;

    putw(hari,pointer_ke_file);
    putw(bulan,pointer_ke_file);
    putw(tahun,pointer_ke_file);
    putw(jam,pointer_ke_file);
    putw(menit,pointer_ke_file);
    putw(detik,pointer_ke_file);
    putw(tmp_awal,pointer_ke_file);
    putw(berat_awal,pointer_ke_file);
    putw(tmp_akhir,pointer_ke_file);
    putw(berat_akhir,pointer_ke_file);
    putw(moisture,pointer_ke_file);
    fclose(pointer_ke_file);
}
```

```

void baca_isi_file()
{
    int    tmp_awall,tmp_akhirr;
    float  berat_awall,berat_akhirr,moisturee;
```

```

int i;

frame(19,9,57,23,113,0,LEFT_UP3,RIGTH_UP3,LEFT_DWN3,RIGTH_DWN3,
      HORIZONTAL2,VERTICAL1);
for(i=1;i<=13;i++){
write_string(20,(9+i),"",112);
write_string(20,10,"    Nama File           :",112);
write_string(20,11,"    Tgl/bln/thn           :",112);
write_string(20,12,"    Jam                   :",112);
write_string(19,13,"|-----|",113);
write_string(20,15,"    Temperatur Awal      :",112);
write_string(20,16,"    Berat Awal          :",112);
write_string(20,18,"    Temperatur Akhir     :",112);
write_string(20,19,"    Berat Akhir          :",112);
write_string(20,21,"    Moisture             :",112);

hari=getw(pointer_ke_file);
bulan=getw(pointer_ke_file);
tahun=getw(pointer_ke_file);
jam=getw(pointer_ke_file);
menit=getw(pointer_ke_file);
detik=getw(pointer_ke_file);

tmp_awall=getw(pointer_ke_file);
berat_awall=getw(pointer_ke_file);
tmp_akhirr=getw(pointer_ke_file);
berat_akhirr=getw(pointer_ke_file);
moisturee=getw(pointer_ke_file);

goxy(43,10);puts(nama_file);
goxy(42,11);printf("%2d/%2d/%4d",hari,bulan,tahun);
goxy(42,12);printf("%2d:%02d:%02d",jam,menit,detik);
goxy(42,15);printf("%-6d %cC",tmp_awall,248);
goxy(42,16);printf("%3.1f",berat_awall);
goxy(42,18);printf("%-6d %cC",tmp_akhirr,248);
goxy(42,19);printf("%3.1f",berat_akhirr);
goxy(42,21);printf("%3.1f",moisturee);

fclose(pointer_ke_file);
}

int file()
{
    char    file_scr[20*5*2],
           baca_scr[35*20*2];
    int     need;

    nama_file=(char *)malloc(20);

```

```

frame(9,3,19,6,23,0,194,194,LEFT_DWN1,RIGTH_DWN1,
      HORIZONTAL1,VERTICAL1);

write_string(10,4,"  ave    ",30);
write_string(10,5,"  oad    ",30);
write_string(11,4,"S",27);
write_string(11,5,"L",27);
paint(1,25,80,25,17,0);
write_string(2,25,"Esc",27);
write_string(5,25,"-to Main Menu",30);
do
{
    need=go_file();
    switch(need)
    {
        case 1:gettext(10,6,37,8,file_scr);
                masukan_nama_file(nama_file);
                kursor(7,6);
                goxy(79,26);
                if(open_file_write(nama_file)) {
                    paint(1,25,80,25,17,0);
                    write_string(2,25,"Saving...",30);
                    rekam_data(); }

                else {
                    paint(1,25,80,25,17,0);
                    write_string(2,25,
                                "Cannot open input file...",30);
                    getch();}
                paint(1,25,80,25,17,0);
                write_string(2,25,"Esc",27);
                write_string(5,25,"-to Main Menu",30);
                kursor(6,7);
                goxy(79,26);
                puttext(10,6,37,8,file_scr);break;
        case 2:gettext(10,6,37,8,file_scr);
                masukan_nama_file(nama_file);
                kursor(7,6);
                goxy(79,26);
                if(open_file_read(nama_file)) {
                    paint(1,25,80,25,17,0);
                    write_string(2,25,"Loading...",30);
                    puttext(10,6,37,8,file_scr);
                    gettext(19,9,58,24,baca_scr);
                    baca_isi_file();
                    goxy(79,26);
                    paint(1,25,80,25,17,0);
                    write_string(2,25,
                                "Press any key to continue...",30);
                    getch();

```

```

        puttext(19,9,58,24,baca_scr);
    }
    else {
        paint(1,25,80,25,17,0);
        write_string(2,25,
            "Cannot open output file...",30);
        getch();
        puttext(10,6,37,8,file_scr);}
    paint(1,25,80,25,17,0);
    write_string(2,25,"Esc",27);
    write_string(5,25,"-to Main Menu",30);
    kursor(6,7);
    goxy(79,26);
    break;
}
}while(need!=0);
}

/*----- INITIAL -----*/
int go_i()
{
    char puzh;
    int choise_i;

    choise_i=var_choise_i;
    hl(23,choise_i+3,94,22);

    do
    {
        puzh=getch();
        hl(23,var_choise_i+3,27,2);
        hl(25,var_choise_i+3,30,20);
        switch(puzh)
        {
            case DOWN :choise_i++;if(choise_i>5)choise_i=1;
                        var_choise_i=choise_i;break;
            case UP   :choise_i--;if(choise_i<1)choise_i=5;
                        var_choise_i=choise_i;break;
            case 'T'  :choise_i=1;var_choise_i=1;puzh=ENTER;break;
            case 't'  :choise_i=1;var_choise_i=1;puzh=ENTER;break;
            case 'K'  :choise_i=2;var_choise_i=2;puzh=ENTER;break;
            case 'k'  :choise_i=2;var_choise_i=2;puzh=ENTER;break;
            case 'p'  :choise_i=3;var_choise_i=3;puzh=ENTER;break;
            case 'E'  :choise_i=4;var_choise_i=4;puzh=ENTER;break;
            case 'e'  :choise_i=4;var_choise_i=4;puzh=ENTER;break;
            case 'Q'  :choise_i=5;var_choise_i=5;puzh=ENTER;break;
            case 'q'  :choise_i=5;var_choise_i=5;puzh=ENTER;break;
            case ESC  :choise_i=0;puzh=ENTER;break;
        }
    }
}

```

```

    }
    hl(23,var_choise_i+3,94,22);
}
while(puzh!=ENTER);
return(choise_i);
}

/*----- Membaca Temperatur Ruang Pemanas -----*/
void temp_awal()
{
    float celcius;
    int i,temp;

    frame(15,5,37,7,113,0,LEFT_UP1,RIGTH_UP1,LEFT_DWN1,RIGTH_DWN1,
        HORIZONTAL1,VERTICAL1);
    write_string(18,5," Furnace Temp.",113);
    write_string(16,6," ",112);
    do{
        while(!kbhit()){
            celcius = konversi2();
            temp=celcius;
            goxy(17,6);
            kursor(7,6);
            printf(" %3d %cC",temp,248);}
        while(getch()!=ESC);
        kursor(6,7);
        goxy(79,26);
    }

/*----- Kalibrasi Timbangan -----*/
void kalibrasi()
{
    float berat;
    int i;

    frame(15,6,37,8,113,0,LEFT_UP1,RIGTH_UP1,LEFT_DWN1,RIGTH_DWN1,
        HORIZONTAL1,VERTICAL1);
    write_string(22,6," Berat ",113);
    write_string(16,7," ",112);
    do{
        while(!kbhit()){
            berat = konversi3();
            goxy(17,7);
            kursor(7,6);
            printf("%4.1f gram ",berat);
        }
    }while(getch()!=ESC);
}

```



```

    kursor(6,7);
    goxy(79,26);
}

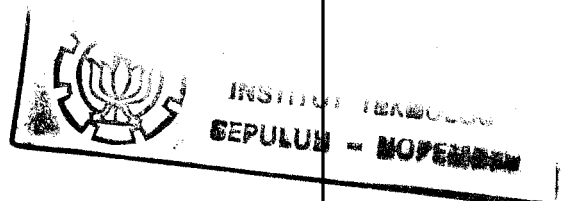
/*----- Tabel -----*/
void tabel()
{
    int i,j=1,count=1;
    float berat;

    frame(46,4,79,24,113,0,LEFT_UP1,RIGTH_UP1,LEFT_DWN1,RIGTH_DWN1,
        HORIZONTAL1,VERTICAL1);
    for(i=0;i<19;i++){
        write_string(47,5+i,"
    write_string(47,5," No.
    write_string(47,6,"
    do{
        while(!kbhit()){
            berat = konversi3();
            kursor(7,6);
            goxy(49,6+j);
            printf("%d",count);
            goxy(67,6+j);
            printf("%4.1f",berat);
            count++;j++;
            if(count>17) {scroll(48,6,77,22,112);j=17;}
        }
        }while(getch()!=ESC);
        kursor(6,7);
        goxy(79,26);
    }

    }

/*----- kontrol kipas penghisap -----*/
void kipas()
{
    if(on_off==ON){
        outportb(PA2,0x01);
        write_string(39,7,"OFF",30);
        on_off=OFF;}
    else{
        outportb(PA2,0x00);
        write_string(39,7,"ON ",30);
        on_off=ON;}
}

```



```

int initial()
{
    char temp0_scr[25*5*2],
        kali_scr[25*5*2],
        ukur_scr[40*40*2];
    int quit=0,need;

    frame(22,3,45,9,23,0,194,194,LEFT_DWN1,RIGTH_DWN1,
        HORIZONTAL1,VERTICAL1);
    write_string(23,4,"emperatur",30);
    write_string(23,5,"alibrasi",30);
    write_string(23,6,"engukuran",30);
    write_string(23,7,"xhaust Fan",30);
    write_string(23,8,"uit",30);
    write_string(24,4,"T",27);
    write_string(24,5,"K",27);
    write_string(24,6,"P",27);
    write_string(24,7,"E",27);
    write_string(24,8,"Q",27);
    if(on_off==ON){
        write_string(39,7,"ON",30);}
    else{
        write_string(39,7,"OFF",30);}
    paint(1,25,80,25,17,0);
    write_string(2,25,"Esc",27);
    write_string(5,25,"-to Main Menu",30);
    do
    {
        need=go_i();
        switch(need)
        {
            case 1:gettext(15,5,38,8,temp0_scr);temp_awal();
                puttext(15,5,38,8,temp0_scr);break;
            case 2:gettext(15,6,38,8,kali_scr);kalibrasi();
                puttext(15,6,38,8,kali_scr);break;
            case 3:gettext(46,4,80,25,ukur_scr);tabel();
                puttext(46,4,80,25,ukur_scr);break;
            case 4:kipas();break;
            case 5:quit=1;outportb(PA2,0x01);need=0;
        }
    }while(need!=0);
    return(quit);
}

/*----- PROCESS -----*/
int go_p()
{

```

```

char puzh;
int choise_p;

choise_p=var_choise_p;
hl(39,choise_p+3,94,9);
do
{
    puzh=getch();
    hl(39,var_choise_p+3,27,2);
    hl(41,var_choise_p+3,30,7);
    switch(puzh)
    {
        case DOWN :choise_p++;if(choise_p>3) choise_p=1;
                    var_choise_p=choise_p;break;
        case UP    :choise_p--;if(choise_p<1) choise_p=3;
                    var_choise_p=choise_p;break;
        case 'S'   :choise_p=1;var_choise_p=1;puzh=ENTER;break;
        case 's'   :choise_p=1;var_choise_p=1;puzh=ENTER;break;
        case 'C'   :choise_p=2;var_choise_p=2;puzh=ENTER;break;
        case 'c'   :choise_p=2;var_choise_p=2;puzh=ENTER;break;
        case 'G'   :choise_p=3;var_choise_p=3;puzh=ENTER;break;
        case 'g'   :choise_p=3;var_choise_p=3;puzh=ENTER;break;
        case ESC   :choise_p=0;puzh=ENTER;break;
    }
    hl(39,var_choise_p+3,94,9);
}
while(puzh!=ENTER);
return(choise_p);
}

int go_s()
{
    char puzh;
    int choise_s;

    choise_s=var_choise_s;
    hl(41,choise_s+5,94,24);
    do
    {
        puzh=getch();
        hl(41,var_choise_s+5,30,24);
        switch(puzh)
        {
            case DOWN :choise_s++;if(choise_s>4) choise_s=1;
                        var_choise_s=choise_s;break;
            case UP    :choise_s--;if(choise_s<1) choise_s=4;
                        var_choise_s=choise_s;break;
            case ESC   :choise_s=0;puzh=ENTER;break;
        }
    }
}

```

```

    }
    hl(41,var_choise_s+5,94,24);
}
while(puzh!=ENTER);
return(choise_s);
}

```

```

int indigit(int x,int y,int len)
{
    int j=0,p=0;
    int var;
    char ch1,ch2,key[1];

    union inkey{
        char ch[2];
        int i;
    }c;

    sprintf(key,"%c",ENTER);
    do
    {
        while(j<len)
        {
            goxy(x+j,y);
            c.i=bioskey(0);
            if(!strchr(key,c.ch[0]))
            {
                if(isdigit(c.ch[0])){
                    /*space(x,y,len);*/
                    p=10*p+c.ch[0]-48;
                    goxy(x,y);printf("%d",p);
                    j++;
                    var=p;}
            }
            else j=len;
        }
    }while((getch()!=ENTER)&&(getch()!=ESC));return(var);
}

```

```

/*----- Setting Time -----*/

```

```

void setwaktu()
{
    int need,puzh;

    frame(42,9,68,13,31,0,LEFT_UP1,RIGTH_UP1,LEFT_DWN1,RIGTH_DWN1,
        HORIZONTAL1,VERTICAL1);
    write_string(43,10,"          Menit :          ",23);
}

```

```

write_string(43,11,"      Jam      :      ",23);
write_string(43,12,"      Detik     :      ",23);

gettime(&t);
goxy(59,10); printf("%2d",t.ti_min);
goxy(59,11); printf("%2d",t.ti_hour);
goxy(59,12); printf("%2d",t.ti_sec);
goxy(61,10);
do{
    puzh=getch();
    switch(puzh)
    {
        case ENTER:need=0;break;
        case ESC  :need=0;break;
        default   :goxy(59,10);printf("      ");
                    t.ti_min=indigit(59,10,2);
                    goxy(59,11);printf("      ");
                    t.ti_hour=indigit(59,11,2);
                    goxy(59,12);printf("      ");
                    t.ti_sec=indigit(59,12,2);
                    settime(&t);need=0;
                    goxy(79,26);break;
    }
}while(need!=0);
goxy(79,26);
}

/*----- Setting Date -----*/
void settanggal()
{
    int need,puzh;

    frame(42,10,68,14,31,0,LEFT_UP1,RIGTH_UP1,LEFT_DWN1,RIGTH_DWN1,
          HORIZONTAL1,VERTICAL1);
    write_string(43,11,"      Tahun      :      ",23);
    write_string(43,12,"      Tanggal   :      ",23);
    write_string(43,13,"      Bulan      :      ",23);

    getdate(&d);
    goxy(59,11); printf("%4d",d.da_year);
    goxy(61,12); printf("%2d",d.da_day);
    goxy(61,13); printf("%2d",d.da_mon);
    goxy(63,11);
    do{
        puzh=getch();
        switch(puzh)
        {
            case ENTER:need=0;break;

```

```

        case ESC :need=0;break;
        default  :goxy(59,11);printf("      ");
                  d.da_year=indigit(59,11,4);
                  goxy(61,12);printf("      ");
                  d.da_day=indigit(61,12,2);
                  goxy(61,13);printf("      ");
                  d.da_mon=indigit(61,13,2);
                  setdate(&d);need=0;
                  goxy(79,26);break;
    }
    }while(need!=0);
    goxy(79,26);
}

/*----- Setting Temperatur -----*/
void settemp()
{
    int need=1,puzh;

    frame(42,7,73,9,31,0,LEFT_UP1,RIGTH_UP1,LEFT_DWN1,RIGTH_DWN1,
          HORIZONTAL1,VERTICAL1);
    write_string(43,8,"  Temperatur      :      xC ",23);
    goxy(62,8);printf(" %3d",setting1);
    do{
        puzh=getch();
        switch(puzh)
        {
            case ENTER:need=0;break;
            case ESC  :need=0;break;
            default   :goxy(62,8);printf("      ");
                      setting1=indigit(62,8,3);need=0;
                      goxy(79,26);setting=setting1;break;
        }
    }while(need!=0);
    goxy(79,26);
}

/*----- Setting Konstanta Proposional -----*/
void setkp()
{
    int need=1,puzh;

    frame(42,8,73,10,31,0,LEFT_UP1,RIGTH_UP1,LEFT_DWN1,RIGTH_DWN1,
          HORIZONTAL1,VERTICAL1);
    write_string(43,9,"  Kp      :      ",23);
    goxy(54,9);printf(" %4.1f",kp);
}

```

```

do{
    puzh=getch();
    switch(puzh)
    {
        case ENTER:need=0;break;
        case ESC :need=0;break;
        default :goxy(54,9);printf(" ");
                goxy(54,9);
                textcolor(WHITE);textbackground(BLUE);
                scanf("%f",&kp);need=0;break;
    }
}while(need!=0);
goxy(79,26);
}

/*----- Kontrol Pemanas -----*/
int feedback(float x)
{
    unsigned PWM;
    float celcius,salah;

    celcius=x;

    salah=setting1-celcius;
    if(salah<0) PWM=(255*(1-exp(kp*salah)));
    if(salah==0) PWM=(255-(kp*setting));
    if(salah>0) PWM=((255-(kp*setting))+(255.*exp(-abs(kp*salah))));
    if(PWM>255) PWM=255;
    return(PWM);
}

/*----- Proses -----*/
void goproses()
{
    unsigned PWM;
    unsigned char sampel_menit,sampel_detik;
    int i,j,tetap=0;
    float berat,temp;
    struct time t;

    frame(1,4,80,24,113,0,LEFT_UP3,RIGTH_UP3,LEFT_DWN3,RIGTH_DWN3,
        HORIZONTAL2,VERTICAL1);
    for(i=1;i<=19;i++){
        for(j=2;j<=79;j++){
            write_string(j,(4+i)," ",112);}
        }
    write_string(1,18,"C",113);write_string(80,18,"4",113);

```

```

for(j=2;j<=79;j++){write_string(j,18,"D",113);}
write_string(34,4, " Monitoring ",113);
write_string(36,18," Kontrol ",113);
write_string(5,6, "Tgl/bln/thn      :",112);
write_string(5,8, "Setting Temperatur :",112);
write_string(12,10,
    "                          Furnace Temperatur :",112);
write_string(12,12,
    "Berat Awal   ( pada temperatur =      ) :",112);
write_string(12,14,
    "Berat Akhir  ( pada temperatur =      ) :",112);
write_string(12,16,
    "                          Moisture :",112);
write_string(5,20,"Exhaust Fan",112);
write_string(5,21,"Kp",112);write_string(20,21,"      ",113);
write_string(5,22,"PWM",112);write_string(20,22,"      ",113);

paint(1,25,80,25,17,0);
write_string(2,25,"Working...",30);

kursor(7,6);
getdate(&d);

goxy(26,6);printf("%2d/%2d/%4d",d.da_day,d.da_mon,d.da_year);
goxy(26,8);printf("%3d %cC",setting1,248);
outportb(PA2,0x01);on_off=OFF;
write_string(20,20,"OFF",113);
goxy(20,21);printf("%3.1f",kp);

tmp_awal=konversi2();
berat_awal=konversi3();

goxy(57,10);printf("%-3d %cC ",tmp_awal,248);
goxy(45,12);printf("%3d %cC ",tmp_awal,248);
goxy(57,12);printf("%4.1f gram ",berat_awal);
do{
    while(!kbhit()){
        temp=konversi2();
        tmp_akhir=temp;
        PWM=feedback(temp);
        outportb(COUNT1,PWM);
        goxy(57,10);printf("%-3d %cC ",tmp_akhir,248);
        goxy(45,14);printf("%3d %cC ",tmp_akhir,248);
        goxy(20,22);printf("%-5d",PWM);
        if(tmp_akhir<106){
            gettime(&t);
            sampel_menit=t.ti_min+30;
            sampel_detik=t.ti_sec;
            if(sampel_menit>=60) sampel_menit=sampel_menit-60;

```



```

        berat_akhir=berat_awal;}

else {
    gettimeofday(&t);
    if((t.ti_min==sampel_menit)&&
        (t.ti_sec>=sampel_detik)){
        berat_akhir=konversi3();
        gettimeofday(&t);
        sampel_menit=t.ti_min+30;
        sampel_detik=t.ti_sec;
        if(sampel_menit>=60) sampel_menit=sampel_menit-60;
        if((oldberat<berat_akhir+0.5)!!
            (old_berat>berat_akhir-0.5)) {tetap=1;}
    }
}
goxy(57,14);printf("%4.1f gram ",berat_akhir);
if(berat_awal!=0.){
    moisture=((berat_awal-berat_akhir)/berat_awal)*100;
    goxy(57,16);printf("%4.1f % ",moisture);
}
if(tetap!=0){
    paint(1,25,80,25,17,0);
    write_string(2,25,"Proses Selesai...",30);
    outportb(COUNT1,0);
    getch();
    break;
}
}
if(tetap!=0) break;
}while(getch()!=ESC);
kursor(6,7);
goxy(79,26);
paint(1,25,80,25,17,0);
write_string(2,25,"Esc",27);
write_string(5,25,"-to Main Menu",30);
outportb(COUNT1,0);
}

/*----- Printing -----*/
void file_name(char *nama_file)
{
    frame(40,6,66,8,31,0,LEFT_UP1,RIGTH_UP1,LEFT_DWN1,RIGTH_DWN1,
        HORIZONTAL1,VERTICAL1);
    paint(1,25,80,25,17,0);
    write_string(2,25,"Esc",27);
    write_string(5,25,"-to Main Menu",30);

    write_string(41,7," Nama File : ",31);
    goxy(55,7);

```

```

textbackground(BLUE);
textcolor(WHITE);
while((*nama_file=getche())!='\r') {++nama_file;}
*(nama_file)='\0';
}

```

```

void isi_file()
{

```

```

    int    tmp_awall,tmp_akhirr;
    float  berat_awall,berat_akhirr,moisturee;

```

```

    hari=getw(pointer_ke_file);
    bulan=getw(pointer_ke_file);
    tahun=getw(pointer_ke_file);
    jam=getw(pointer_ke_file);
    menit=getw(pointer_ke_file);
    detik=getw(pointer_ke_file);
    tmp_awall=getw(pointer_ke_file);
    berat_awall=getw(pointer_ke_file);
    tmp_akhirr=getw(pointer_ke_file);
    berat_akhirr=getw(pointer_ke_file);
    moisturee=getw(pointer_ke_file);

```

```

    fclose(pointer_ke_file);

```

```

    if(status()==144){

```

```

        paint(1,25,80,25,17,0);

```

```

        write_string(2,25,"Printing...",30);

```

```

        fprintf(stdprn, "\x1B\x7");

```

```

        fputs("===== MOISTURE: SOLVER =====",stdprn);

```

```

        fputs("Nama File : ",stdprn);

```

```

        fputs(nama_file,stdprn);

```

```

        fputs("\n",stdprn);

```

```

        fputs("Tgl/bln/thn : ",stdprn);

```

```

        fprintf(stdprn,"%2d/%2d/%4d",hari,bulan,tahun);

```

```

        fputs("\n",stdprn);

```

```

        fputs("Jam : ",stdprn);

```

```

        fprintf(stdprn,"%2d:%02d:%02d",jam,menit,detik);

```

```

        fputs("\n",stdprn);

```

```

        fputs("===== ",stdprn);

```

```

        fputs("===== ",stdprn);

```

```

        fputs("Temperatur Awal : ",stdprn);

```

```

        fprintf(stdprn,"%-3d %cC",tmp_awall,248);

```

```

        fputs("\n",stdprn);

```

```

        fputs("Berat Awal : ",stdprn);

```

```

        fprintf(stdprn,"%4.1f gram",berat_awall);

```

```

        fputs("\n",stdprn);

```

```

fputs("|\n",stdprn);
fputs("    Temperatur Akhir : ",stdprn);
fprintf(stdprn,"%-3d  %cC",tmp_akhirr,248);
fputs("|\n",stdprn);
fputs("    Berat Akhir      : ",stdprn);
fprintf(stdprn,"%4.1f  gram",berat_akhirr);
fputs("|\n",stdprn);
fputs("|\n",stdprn);
fputs("    Moisture          : ",stdprn);
fprintf(stdprn,"%4.1f  %",moisturee);
fputs("|\n",stdprn);
fputs("|\n",stdprn);
fputs("|\n",stdprn);
fputs("\x0c",stdprn);
fflush(stdprn); }
else {
    paint(1,25,80,25,17,0);
    write_string(2,25,"Error accessing PRN device...",30);
    getch();
}

}

void cetak()
{
    char cetak_scr[20*5*2];

    gettext(40,6,67,8,cetak_scr);
    file_name(nama_file);
    goxy(79,26);
    if(open_file_read(nama_file)) {
        paint(1,25,80,25,17,0);
        write_string(2,25,"Loading...",30);
        isi_file();
        puttext(40,6,67,8,cetak_scr);}
    else {
        paint(1,25,80,25,17,0);
        write_string(2,25,"Cannot open output file...",30);
        getch();
        puttext(40,6,67,8,cetak_scr);}
    paint(1,25,80,25,17,0);
    write_string(2,25,"Esc",27);
    write_string(5,25,"-to Main Menu",30);
    goxy(79,26);
}

```

```

/*----- Menu Setting -----*/
void menu_set()
{
    char fur_scr[35*15*2],
        kp_scr[35*15*2],
        time_scr[35*15*2],
        set_scr[35*15*2],
        date_scr[35*15*2];
    int need;

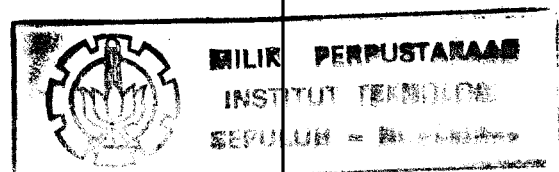
    gettext(40,5,74,12,set_scr);
    frame(40,5,65,10,23,0,LEFT_UP1,RIGTH_UP1,LEFT_DWN1,RIGTH_DWN1,
        HORIZONTAL1,VERTICAL1);
    write_string(41,6," Furnace Temperatur ",30);
    write_string(41,7," Konstanta Proposional ",30);
    write_string(41,8," Waktu ",30);
    write_string(41,9," Tanggal ",30);
    do
    {
        need=go_s();
        switch(need)
        {
            case 1: gettext(40,6,74,10,fur_scr); settemp();
                    puttext(40,6,74,10,fur_scr); break;
            case 2: gettext(40,6,74,11,kp_scr); setkp();
                    puttext(40,6,74,11,kp_scr); break;
            case 3: gettext(40,6,70,13,time_scr); setwaktu();
                    puttext(40,6,70,13,time_scr); break;
            case 4: gettext(40,6,70,14,date_scr); settanggal();
                    puttext(40,6,70,14,date_scr); break;
        }
    }while(need!=0);
    puttext(40,5,74,12,set_scr);
}

/*----- Menu Proses -----*/
int process()
{
    int need,pilih=0;

    nama_file=(char *)malloc(20);

    frame(38,3,48,7,23,0,194,194,LEFT_DWN1,RIGTH_DWN1,
        HORIZONTAL1,VERTICAL1);
    write_string(39,4," etting ",30);
    write_string(39,5," etak ",30);
    write_string(39,6," o ",30);

```



```

write_string(40,4,"S",27);
write_string(40,5,"C",27);
write_string(40,6,"G",27);
paint(1,25,80,25,17,0);
write_string(2,25,"Esc",27);
write_string(5,25,"-to Main Menu",30);
do
{
    need=go_p();
    switch(need)
    {
        case 1:menu_set();break;
        case 2:cetak();break;
        case 3:pilih=1;break;
    }
}while((need!=0)&&(pilih!=1));
return(pilih);
}

/*----- HELP -----*/
help()
{
    int a;

    gettext(13,5,66,23,help_scr);
    frame(13,6,66,22,31,0,LEFT_UP2,RIGTH_UP2,LEFT_DWN2,RIGTH_DWN2,
        HORIZONTAL2,VERTICAL2);
    for(a=0;a<=14;a++){
        write_string(14,a+7,text[a],31);}

    paint(1,25,80,25,17,0);
    write_string(2,25,"Esc",27);
    write_string(5,25,"-to Main Menu",30);
    while(getch()!=ESC);
    puttext(13,5,66,23,help_scr);
}

/*----- MAIN PROGRAM -----*/
int go_main()
{
    char puzh;

    paint(1,25,80,25,17,0);
    write_string(60,25,"@ Karang-TE-FTI-ITS",30);
    write_string(58,25,"3",27);
    hl(choise+2,2,94,lng);

```

```

do
{
    puzh=getch();
    hl(choise+2,2,27,1);
    hl(choise+3,2,30,lng);
    switch(puzh)
    {
        case RIGHT:var_choise++;if(var_choise>5) var_choise=1;break;
        case LEFT :var_choise--;if(var_choise<1) var_choise=5;break;
        case 'F' :var_choise=2;puzh=ENTER;break;
        case 'f' :var_choise=2;puzh=ENTER;break;
        case 'I' :var_choise=3;puzh=ENTER;break;
        case 'i' :var_choise=3;puzh=ENTER;break;
        case 'P' :var_choise=4;puzh=ENTER;break;
        case 'p' :var_choise=4;puzh=ENTER;break;
        case 'H' :var_choise=5;puzh=ENTER;break;
        case 'h' :var_choise=5;puzh=ENTER;break;
    }
    switch(var_choise)
    {
        case 1:choise=1;lng=1;break;
        case 2:choise=8;lng=4;break;
        case 3:choise=21;lng=7;break;
        case 4:choise=37;lng=6;break;
        case 5:choise=53;lng=4;break;
    }
    hl(choise+2,2,94,lng);
}
while(puzh!=ENTER);
return(var_choise);
}

main()
{
    char about1_scr[15*7*2];
    int i,j,puzh,need,pilih;

    outportb(CWR1,0x80); /* Control Word PPI 8255 #1 */
    outportb(CWR2,0x8A); /* Control Word PPI 8255 #2 */
    outportb(CWR3,0x13); /* Control Word PIT 8253 */

    clrscr();
    gettext(1,1,80,25,screen);
    gettext(1,25,80,25,fastkey_scr);
    gettext(1,1,80,3,main_scr);
    frame(1,1,80,3,23,0,LEFT_UP1,RIGTH_UP1,LEFT_DWN1,RIGTH_DWN1,
        HORIZONTAL1,VERTICAL1);
    write_string(29,1,"Moisture:Solver",30);

```

```

paint(2,2,79,2,17,0);
write_string(3,2,"p",30);
write_string(11,2,"ile",30);
write_string(24,2,"nitia",30);
write_string(40,2,"roses",30);
write_string(56,2,"elp",30);
write_string(10,2,"F",27);
write_string(23,2,"I",27);
write_string(38,2,"P",27);
write_string(55,2,"H",27);
goxy(79,26);
do{
    need=go_main();
    switch(need)
    {
        case 1:gettext(1,3,12,7,about1_scr);puzh=about1();
                puttext(1,3,12,7,about1_scr);break;
        case 2:gettext(9,3,21,7,file_scr);file();
                puttext(9,3,21,7,file_scr);break;
        case 3:gettext(22,3,47,9,i_scr);puzh=initial();
                puttext(22,3,47,9,i_scr);break;
        case 4:gettext(38,3,49,8,p_scr);pilih=process();
                puttext(38,3,49,8,p_scr);break;
        case 5:help();break;
    }
    if(pilih==1){
        gettext(1,4,80,24,go_scr);goproses();
        puttext(1,4,80,24,go_scr);
        pilih=0;}
    } while(puzh!=1);
    puttext(1,1,80,3,main_scr);
    puttext(1,25,80,1,fastkey_scr);
    puttext(1,1,80,25,screen);
}

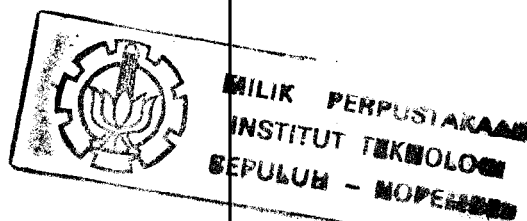
```

```
/* File : PROCEDUR.C */
```

```
/*----- PROCEDUR -----*/
```

```
void goxy(char x, char y)
{
    union REGS regs;

    regs.h.bh=0;
    regs.h.dl=x-1;
    regs.h.dh=y-1;
    regs.h.ah=0x02;
    int86(0x10,&regs,&regs);
}
```



```
void write_string(int x,int y,char *p,int attrib)
{
    register int i;
    char far *v;

    v=(char far*)0xB8000000;
    v+=(x-1)*2+(y-1)*160;
    for (;*p;)
    {
        *v++=*p++;
        *v++=attrib;
    }
}
```

```
void write_char(int x,int y,int ch,int attrib)
{
    char far *v;
    v=(char far*)0xB8000000;
    v+=((x-1)*2+(y-1)*160);
    *v++=ch;
    *v=attrib;
}
```

```
void paint(int x1,int y1,int x2,int y2,int color,int mode)
{
    register int x,y;
    char far *pntptr;

    pntptr=(char far *) 0xB8000000;
    pntptr+=(x1-1)*2+(y1-1)*160;
    for (y=y1;y<=y2;y++)
```



```

    for (x=x1;x<=x2;x++)
    {
        pntptr=(char far *) 0xB8000000+(x-1)*2+(y-1)*160;
        if(mode) pntptr++;
        else *pntptr+=0;
        *pntptr+=color*16;
    }
}

void frame(int x1,int y1,int x2,int y2,int back,int shadow,int
           left_up, int righth_up,int left_dwn,int righth_dwn,int
           horizontal,int vertical)
{
    register int down, hori;
    write_char(x1,y1,left_up,back);
    for (down=y1+1;down<y2;down++)
        write_char(x1,down,vertical,back);
    write_char(x1,y2,left_dwn,back);
    for (hori=x1+1;hori<x2;hori++)
    {
        write_char(hori,y1,horizontal,back);
        write_char(hori,y2,horizontal,back);
    }
    write_char(x2,y1,righth_up,back);
    for (down=y1+1;down<y2;down++)
        write_char(x2,down,vertical,back);
    write_char(x2,y2,righth_dwn,back);
    if(shadow)
    {
        paint(x2+1,y1+1,x2+1,y2+1,7,1);
        paint(x1+1,y2+1,x2,y2+1,7,1);
    }
    paint(x1+1,y1+1,x2-1,y2-1,back,0);
}

void hl(int h,int v, int attribut,int lg)
{
    char far*vidmem;
    int hit;
    vidmem=(char far*) 0xB8000001;
    vidmem+=(h-1)*2+(v-1)*160;
    for (hit=1;hit<=lg;hit++)
        {*vidmem=attribut;vidmem+=2;}
}

```

```
unsigned char sensor(unsigned char channel)
{
    unsigned char EOC,data;
    register int i;

    for(i=0;i<2;i++)
    {
        outportb(PC2,channel);
        outportb(PC2,channel|0x08);
        outportb(PC2,channel);
        do{EOC=inportb(PC2)&0x10;}while(!EOC);
        data=inportb(PB2);
    }
    return(data);
}

int status()
{
    union REGS regs;

    regs.h.ah = 2;
    regs.x.dx = 0;
    int86(0x17,&regs,&regs);
    return(regs.h.ah);
}

void kursor(unsigned char awal, unsigned char akhir)
{
    union REGS regs;

    regs.h.ch=awal;
    regs.h.cl=akhir;
    regs.h.ah=1;
    int86(0x10,&regs,&regs);
}

void scroll(x1,y1,x2,y2,attrib)
unsigned char x1,y1,x2,y2,attrib;
{
    union REGS regs;

    regs.h.al=1;
    regs.h.bh=attrib;
    regs.h.cl=x1;
    regs.h.ch=y1;
    regs.h.dl=x2;
```



```
regs.h.dh=y2;
regs.h.ah=0x06;
int86(0x10,&regs,&regs);
}
```

```
float konversi(unsigned char channel)
{
    unsigned char data;
    float hasil;

    data=sensor(channel);
    hasil=(data*1.28)/255.;
    return(hasil);
}
```

```
float konversi2()
{
    float hasil,temp;

    temp=konversi(0);
    hasil=100.*temp;
    if(hasil<32.) {return(hasil);}
    else {
        hasil=145.0*temp-12.2;
        return(hasil);
    }
}
```

```
float konversi3()
{
    float berat,hasil,temp;
    int temp1;

    temp=konversi2();
    temp1=temp;
    berat=konversi();
    if(temp>106) {hasil=19.855*berat-0.030;}
    else {hasil=20.0*berat-0.133;}
    return(hasil);
}
```

```
/* File : DECLARES.H */
```

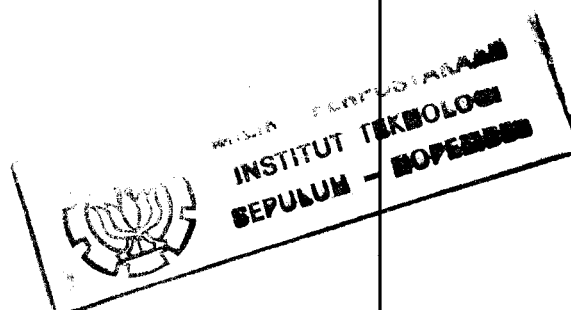
```
#include <ctype.h>
#include <alloc.h>
#include <stdio.h>
#include <stdlib.h>
#include <dos.h>
#include <math.h>
#include <conio.h>
```

```
#define LEFT_UP1 218
#define LEFT_UP2 201
#define LEFT_UP3 213
#define RIGTH_UP1 191
#define RIGTH_UP2 187
#define RIGTH_UP3 184
#define LEFT_DWN1 192
#define LEFT_DWN2 200
#define LEFT_DWN3 212
#define RIGTH_DWN1 217
#define RIGTH_DWN2 188
#define RIGTH_DWN3 190
#define HORIZONTAL1 196
#define HORIZONTAL2 205
#define VERTICAL1 179
#define VERTICAL2 186
#define HOME 71
#define LEFT 75
#define RIGHT 77
#define UP 72
#define DOWN 80
#define ESC 27
#define ENTER 13
#define ON 1
#define OFF 0
```

```
#define PA1 0x300
#define PB1 0x301
#define PC1 0x302
#define CWR1 0x303
```

```
#define PA2 0x304
#define PB2 0x305
#define PC2 0x306
#define CWR2 0x307
```

```
#define COUNT1 0x308
#define COUNT2 0x309
#define COUNT3 0x30A
#define CWR3 0x30B
```



```

struct  date d, time t;
float   setting=107., berat_awal=0., berat_akhir=0.,
        oldberat=0., moisture=0., kp=2.3;
int      setting1=107, setting2, tmp_awal=0, tmp_akhir=0,
        on_off=ON, tahun;
char     hari, bulan, jam, menit,
        detik, *nama_file;

```

```
FILE *pointer_ke_file;
```

```

char *about[8]={
    "
    "  ALAT UNTUK MENENTUKAN KANDUNGAN AIR DALAM BATUBARA  ",
    "          YANG DIINTERFACEKAN PADA IBM PC/XT          ",
    "
    "                      oleh :
    "                      I Komang Widnyana Karang
    "                      Nrp. 2882200945
    "
    };

```

```

char *text[]={
    "
    "  Pengukuran Kandungan Air dalam Batubara dilakukan  ",
    "  dengan dengan metoda 'PENGERINGAN', yaitu dengan   ",
    "  menguapkan air pada permukaan dan yang melekat.    ",
    "  Persamaan :
    "
    "                      (B0 - B1)
    "          K =  $\frac{\quad}{B0}$  x 100%
    "
    "  dimana :
    "
    "  K   : Kandungan air dalam Batubara [%].
    "  B0  : Berat mula-mula [gram].
    "  B1  : Berat akhir setelah dikeringkan [gram].
    "
    };

```

```

char   about2_scr[24*12*2], screen[80*25*2], main_scr[80*3*2],
        fastkey_scr[1*80*2], file_scr[10*5*2], p_scr[15*8*2],
        i_scr[18*8*2], go_scr[80*25*2], mess_scr[25*8*2],
        help_scr[24*20*2];

```

```

int     var_choise_f=1, var_choise_i=1, var_choise_p=1,
        var_choise_s=1, var_choise=1, choise=1,
        lng=1;

```

## USULAN TUGAS AKHIR

1. Judul Tugas Akhir : ALAT UNTUK MENENTUKAN KANDUNGAN AIR  
DALAM BATUBARA YANG DIINTERFACEKAN  
PADA IBM PC/XT.

2. Ruang Lingkup : - Mikroelektronika  
- Rangkaian Linier Aktif  
- Rangkaian Logika  
- Instrument Elektronika

3. Latar Belakang : Seperti diketahui bahwa Indonesia memiliki sejumlah energi yang cukup potensial untuk dikembangkan, diantaranya batubara. Pengembangan sumber daya batubara merupakan kebijaksanaan diversifikasi energi yang mendorong peralihan dari ketergantungan minyak kepada penggunaan sumber daya yang lebih beragam.

Pengkajian gasifikasi batubara adalah salah satu usaha yang dilakukan untuk memanfaatkan batubara untuk menghasilkan gas, terutama gas karbon monoksida dan hidrogen. Kemudian gas tersebut dapat diolah menghasilkan berbagai produk seperti misalnya:

metanol, bahan bakar sintetis, bahan-bahan kimia, urea, atau gas hasil gasifikasi itu dapat dimanfaatkan sebagai pembangkit tenaga listrik.

Jumlah air dalam batu bara mempengaruhi kondisi gasifikasi, yaitu substokiometri (menghitung berat dan jumlah setiap komponen kimia dalam suatu reaksi kimia) oksigen. Agar tidak mengalami kesulitan dalam penentuan kondisi gasifikasi maka perlu menetapkan kandungan air dalam batubara.

Penentuan kandungan air dalam batubara dilakukan dengan cara pengeringan. Pengeringan batubara dengan menguapkan air pada permukaan dan yang melekat. Menentukan kandungan air dalam batubara secara konvensional membutuhkan peran manusia yang cukup besar karena saat pengeringan dilakukan penimbangan berulang-ulang sampai diperoleh berat yang konstan. Hal ini tentu saja kurang efektif.

Pemakaian Komputer IBM PC/XT secara optimal dalam industrialisasi antara lain sebagai pengontrol proses.

Demikian pula halnya dengan proses untuk menentukan kandungan air dalam batubara dimana parameter-parameter berat dan temperatur dapat dikontrol melalui komputer IBM PC/XT sehingga diharapkan mampu menanggulangi permasalahan yang ada.

4. Penelaahan studi : - Mempelajari sifat-sifat dari batubara.
- Mempelajari rangkaian-rangkaian linier aktif.
  - Mempelajari sistem mikroprosesor.
  - Mempelajari bahasa pemrograman level tinggi serta level rendah.

5. Tujuan : Merancang dan membuat alat untuk menentukan kandungan air dalam batubara yang diinterfacekan pada komputer IBM PC/XT.

6. Relevansi : Dengan alat ini dibuat diharapkan dapat bermanfaat bagi industri-industri yang memakai batubara sebagai bahan baku.



7. Langkah - langkah : 1. Studi literatur.
2. Perencanaan dan pembuatan alat.
3. Pengujian alat.
4. Penulisan naskah.

8. Jadwal Kegiatan :

| K E G I A T A N                   | BULAN KE- |    |    |    |    |    |
|-----------------------------------|-----------|----|----|----|----|----|
|                                   | 1         | 2  | 3  | 4  | 5  | 6  |
| 1. Studi literatur                | ██        | ██ | ██ |    |    |    |
| 2. Perencanaan dan pembuatan alat |           | ██ | ██ | ██ |    |    |
| 3. Pengujian alat                 |           |    |    |    | ██ |    |
| 4. Penulisan naskah               |           |    |    |    | ██ | ██ |

## RIWAYAT HIDUP



I KOMANG WIDNYANA KARANG, dilahirkan di Amlapura, Kabupaten Karangasem Bali pada tanggal 4 Juli 1970 dan merupakan putra ketiga dari tiga bersaudara dari Bapak I Ketut Sulendra dan Ibu Ni Wayan Suati.

Terdaftar sebagai mahasiswa Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya pada tahun 1988 dengan nomor register pokok 2882200945.

Pendidikan yang pernah ditempuh selama ini adalah :

- SD No. 5 Karangasem, kab. Karangasem, 1976-1982
- SMP Neg. 2 Amlapura, Kab. Karangasem, 1982-1985
- SMA Neg. 1 Karangasem, Kab. Karangasem, 1985-1986
- SMA Neg. 1 Singaraja, Kab. Buleleng, 1986-1988
- Jurusan Teknik Elektro FTI - ITS Surabaya 1988

Selama menjadi mahasiswa pernah menjadi asisten praktikum Rangkaian Listrik, asisten praktikum Elektronika dan asisten praktikum Elektronika Lanjutan II.